

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Nina Krmac

POSTAVITEV NOVE ARHITEKTURE SISTEMOV ZA PODPORO  
POSLOVANJU V TELEKOMUNIKACIJSKEM PODJETJU

MAGISTRSKO DELO

Mentor: izr. prof. dr. Marjan Krisper

Ljubljana, 2016

Rezultati magistrskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljane ali izkoriščanje rezultatov magistrskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.



Številka: 146-MAG-ISO/2016

Datum: 29. 02. 2016

**Nina KRMAC**, univ. dipl. inž. rač. in inf.

**L j u b l j a n a**

Fakulteta za računalništvo in informatiko Univerze v Ljubljani izdaja naslednjo magistrsko nalogo

Naslov naloge: **Postavitev nove arhitekture sistemov za podporo poslovanju v telekomunikacijskem podjetju**

**A new architecture setup for business support systems in a telecommunications company**

Tematika naloge:

V magistrski nalogi opišite postopek prenove arhitekture sistemov za podporo poslovanju v telekomunikacijskem podjetju in jo umestite v širši okvir poslovno-informacijske arhitekture. Izberite metodologijo, ki jo boste uporabili za postopek prenove. Opišite jo ter zagovarjajte svojo izbiro. Opišite funkcijo dela arhitekture, ki se prenavlja, opredelite motivacijo za prenovo in pričakovane koristi, ki naj bi jih prenova prinesla. Cilji prenove naj bodo jasno začrtani in ovrednoteni.

Na podlagi ugotovljenih ciljev pripravite konkreten primer predloga prenove ter predstavite, kako predlagana ciljna arhitektura izpolnjuje zadane cilje. Vključite tako prenovo poslovnih procesov kot informacijskih sistemov. Vsako spremembo natančno opišite, zagovarjajte svojo izbiro ter jo predstavite tako v pisni kot grafični obliki. Za modeliranje v grafični obliki izberite primeren modelirni jezik.

Mentor:

izr. prof. dr. Marjan Krisper



Dekan:

prof. dr. Nikolaj Zimic



# Zahvala

Zahvaljujem se mentorju, izr. prof. dr. Marjanu Krisperju, za vse nasvete in pomoč, ne le ob izdelavi magistrskega dela, temveč tudi tekom celotnega študija.

Posebna zahvala gre mojim najbližjim. Valu in Roku za potrpežljivost ter staršem, ki so me že od malega podpirali na moji akademski poti.



# Kazalo

|   |           |
|---|-----------|
| <b>POVZETEK.....</b>  | <b>1</b>  |
| <b>ABSTRACT .....</b>   | <b>3</b>  |
| <b>1 UVOD .....</b>   | <b>5</b>  |
| <b>2 SISTEMI ZA PODPORO POSLOVANJU V TELEKOMUNIKACIJAH.....</b>                     | <b>7</b>  |
| <b>3 POSLOVNO-INFORMACIJSKA ARHITEKTURA .....</b>                                   | <b>11</b> |
| <b>4 TOGAF 9.1.....</b>   | <b>17</b> |
| 4.1 UVOD V TOGAF .....  | 17        |
| 4.2 PRELIMINARNA FAZA .....   | 20        |
| 4.3 FAZA A: ARHITEKTURNA VIZIJA .....   | 21        |
| 4.4 FAZE B, C IN D: POSLOVNA, INFORMACIJSKA IN TEHNOLOŠKA ARHITEKTURA.....          | 21        |
| 4.4.1 Faza B: poslovna arhitektura .....  | 21        |
| 4.4.2 Faza C: informacijska arhitektura .....                                       | 22        |
| 4.4.3 Faza D: tehnološka arhitektura.....   | 22        |
| 4.5 FAZE E DO H: REŠITVE, PLAN MIGRACIJE, IMPLEMENTACIJA, UPRAVLJANJE S SPREMEMBAMI | 23        |
| 4.5.1 Faza E: priložnosti in rešitve .....  | 23        |
| 4.5.2 Faza F: planiranje migracije .....  | 23        |
| 4.5.3 Faza G: upravljanje implementacije .....                                      | 23        |
| 4.5.4 Faza H: upravljanje s spremembami.....  | 23        |

|           |  |           |
|-----------|--|-----------|
| 4.6       | TOGAF DOKUMENT .....                                 | 24        |
| <b>5</b>  | <b>ARCHIMATE 2.1 .....</b>                           | <b>25</b> |
| 5.1       | STRUKTURA.....                                       | 26        |
| 5.2       | RAZŠIRITVE .....                                     | 29        |
| 5.3       | SOUPORABA S TOGAF-OM .....                           | 30        |
| <b>6</b>  | <b>FRAMEWORK.....</b>                                | <b>35</b> |
| <b>7</b>  | <b>ARHITEKTURNE DOBRE PRAKSE .....</b>               | <b>39</b> |
| <b>8</b>  | <b>PRELIMINARNA FAZA IN ARHITEKTURNA VIZIJA.....</b> | <b>41</b> |
| <b>9</b>  | <b>POSLOVNA ARHITEKTURA .....</b>                    | <b>45</b> |
| 9.1       | TRENTNA POSLOVNA ARHITEKTURA .....                   | 49        |
| 9.2       | CILJNA POSLOVNA ARHITEKTURA.....                     | 52        |
| 9.3       | ANALIZA VRZELI POSLOVNE ARHITEKTURE.....             | 55        |
| <b>10</b> | <b>INFORMACIJSKA ARHITEKTURA .....</b>               | <b>57</b> |
| 10.1      | TRENTNA INFORMACIJSKA ARHITEKTURA.....               | 60        |
| 10.2      | CILJNA INFORMACIJSKA ARHITEKTURA .....               | 68        |
| 10.3      | ANALIZA VRZELI INFORMACIJSKE ARHITEKTURE .....       | 77        |
| <b>11</b> | <b>TEHNOLOŠKA ARHITEKTURA .....</b>                  | <b>81</b> |
| 11.1      | TRENTNA TEHNOLOŠKA ARHITEKTURA.....                  | 82        |
| 11.2      | CILJNA TEHNOLOŠKA ARHITEKTURA .....                  | 85        |
| 11.3      | ANALIZA VRZELI TEHNOLOŠKE ARHITEKTURE .....          | 87        |
| <b>12</b> | <b>IMPLEMENTACIJA ARHITEKTURE .....</b>              | <b>89</b> |
| 12.1      | FAZA E: PRILOŽNOSTI IN REŠITVE.....                  | 89        |
| 12.1.1    | <i>Tranzicijska arhitektura 1 .....</i>              | <i>90</i> |
| 12.1.2    | <i>Tranzicijska arhitektura 2 .....</i>              | <i>93</i> |



|           |   |            |
|-----------|---|------------|
| 12.1.3    | <i>Prehod iz tranzicijske arhitekture 2 na ciljno arhitekturo .....</i> | 95         |
| 12.2      | FAZA F: PLANIRANJE MIGRACIJE .....                                      | 96         |
| 12.3      | FAZA G: UPRAVLJANJE IMPLEMENTACIJE .....                                | 97         |
| <b>13</b> | <b>UPRAVLJANJE SPREMEMB V ARHITEKTURI .....</b>                         | <b>99</b>  |
| <b>14</b> | <b>SKLEPNE UGOTOVITVE .....</b>   | <b>101</b> |
|           | <b>VIRI IN LITERATURA .....</b>   | <b>103</b> |
|           | LITERATURA .....  | 103        |
|           | OSTALI VIRI.....  | 104        |



# Kazalo slik

|   |    |
|---|----|
| Slika 4-1: Cikel razvoja arhitekture po TOGAF-u [11] .....                        | 18 |
| Slika 4-2: Entitete TOGAF vsebinskega meta modela [11] .....                      | 19 |
| Slika 4-3: Povezava med artefakti in meta modelom[11] .....                       | 20 |
| Slika 5-1: Jedro ArchiMate ogrodja [10].....                                      | 27 |
| Slika 5-2: Preslikava ogrodij ArchiMate in TOGAF drugega na drugo[10] .....       | 31 |
| Slika 6-1: Uporaba eTOM, SID in TAM v fazah ADM [6] .....                         | 37 |
| Slika 8-1: Katalog principov .....  | 42 |
| Slika 8-2: Model motivacijske ravni .....   | 44 |
| Slika 9-1: eTOM razdelek 1.1.9 Prodaja [15] .....                                 | 46 |
| Slika 9-2: eTOM razdelek 1.2.7.1 Razvoj in umik produktnih specifikacij [15]..... | 47 |
| Slika 9-3: eTOM razdelek 1.3.3 Ravnanje z naročili [15].....                      | 47 |
| Slika 9-4: eTOM razdelek 1.3.7 Ravnanje s težavami [15].....                      | 48 |
| Slika 9-5: eTOM razdelek 1.3.6 Upravljanje s podatki o stranki [15] .....         | 48 |
| Slika 9-6: Trenutna poslovna arhitektura – 1 .....                                | 51 |
| Slika 9-7: Trenutna poslovna arhitektura – 2 .....                                | 52 |
| Slika 9-8: Ciljna poslovna arhitektura – 1 .....                                  | 53 |
| Slika 9-9: Ciljna poslovna arhitektura – 2 .....                                  | 54 |
| Slika 9-10: Analiza vrzeli poslovne arhitekture .....                             | 55 |

|  |    |
|--|----|
| Slika 10-1: Trenutna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti.....           | 61 |
| Slika 10-2: Diagram trenutne realizacije procesa priprave pogodbe .....                                      | 62 |
| Slika 10-3: Diagram trenutne realizacije procesa priprave aneksa k pogodbi .....                             | 63 |
| Slika 10-4: Diagram trenutne realizacije procesov izvajanja sprememb .....                                   | 63 |
| Slika 10-5: Diagram trenutne realizacije procesov za pripravo pogodb in aneksov preko spletne trgovine ..... | 64 |
| Slika 10-6: Diagram trenutne realizacije procesa za ravnanje s pritožbami .....                              | 65 |
| Slika 10-7: Diagram trenutne realizacije procesa za implementacijo novega produkta .....                     | 65 |
| Slika 10-8: Diagram sodelovanja aplikacij za podpisovanje dokumentov .....                                   | 66 |
| Slika 10-9: Trenutna aplikacijska arhitektura .....  | 67 |
| Slika 10-10: Ciljna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti .....           | 69 |
| Slika 10-11: Diagram ciljne realizacije procesa priprave pogodbe .....                                       | 70 |
| Slika 10-12: Diagram ciljne realizacije procesa priprave aneksa na pogodbo .....                             | 71 |
| Slika 10-13: Diagram ciljne realizacije procesov izvajanja sprememb .....                                    | 72 |
| Slika 10-14: Diagram ciljne realizacije procesov za pripravo pogodb in aneksov preko spletne trgovine .....  | 73 |
| Slika 10-15: Diagram ciljne realizacije procesa za ravnanje s pritožbami .....                               | 73 |
| Slika 10-16: Diagram ciljne realizacije procesa za implementacijo novega produkta .....                      | 74 |
| Slika 10-17: Ciljna aplikacijska arhitektura.....  | 75 |
| Slika 10-18: Analiza vrzeli podatkovne arhitekture .....   | 77 |
| Slika 10-19: Analiza vrzeli aplikacijske arhitekture .....   | 78 |
| Slika 11-1: Trenutna tehnološka arhitektura (dekompozicijski diagram platform) .....                         | 83 |
| Slika 11-2: Trenutna tehnična realizacija aplikacij - 1 .....  | 84 |

|  |    |
|--|----|
| Slika 11-3: Trenutna tehnična realizacija aplikacij – 2 .....                      | 84 |
| Slika 11-4: Ciljna tehnološka arhitektura (dekompozicijski diagram platform) ..... | 85 |
| Slika 11-5: Ciljna tehnološka realizacija aplikacij – 1 .....                      | 86 |
| Slika 11-6: Ciljna realizacija aplikacij – 2 .....                                 | 87 |
| Slika 11-7: Analiza vrzeli tehnološke arhitekture .....                            | 88 |
| Slika 12-1: Informacijska tranzicijska arhitektura 1.....                          | 92 |
| Slika 12-2: Infrastrukturna tranzicijska arhitektura 1 .....                       | 92 |
| Slika 12-3: Poslovna tranzicijska arhitektura 1 .....                              | 93 |
| Slika 12-4: Informacijska tranzicijska arhitektura 2.....                          | 94 |
| Slika 12-5: Infrastrukturna tranzicijska arhitektura 2.....                        | 95 |



# Kazalo tabel

|   |    |
|---|----|
| Tabela 5-1: Preslikava entitet iz skupine "Arhitekturna vizija, principi in zahteve ..... | 32 |
| Tabela 5-2: Preslikava entitet iz skupine "Poslovna arhitektura" .....                    | 33 |
| Tabela 5-3: Preslikava entitet iz skupine "Informacijska arhitektura" .....               | 33 |
| Tabela 5-4: Preslikava entitet iz skupine "Tehnološka arhitektura" .....                  | 34 |





# Seznam kratic

|       |   |
|-------|---|
| BSS   | Business Support Systems (Sistemi za podporo poslovanju)                            |
| CRM   | Customer Relationship Management (Upravljanje z odnosi s strankami)                 |
| ESB   | Enterprise Service Bus (Storitveno vodilo)  |
| eTOM  | Enhanced Telecom Operations Map (Razširjen zemljevid telekomunikacijskih operacij)  |
| IT    | Information Technology (Informacijska tehnologija)                                  |
| OM    | Order Management (Upravljanje z naročili)   |
| OMS   | Order Management System (Sistem za upravljanje z naročili)                          |
| OSS   | Operations Support Systems (Sistemi za podporo )                                    |
| PC    | Product catalog (Produktni katalog)   |
| SID   | Shared Information/Data Model (Skupni informacijski/podatkovni model)               |
| SLA   | Service Level Agreement (Dogovor o nivoju podpore)                                  |
| TAM   | Telecom Application Map (Zemljevid aplikacij telekomunikacijskega podjetja)         |
| TOGAF | The Open Group Architecture Framework (Arhitekturno ogrodje skupine The Open Group) |



## Povzetek

Telekomunikacije so bile še do nedavnega cvetoča panoga v naglem razvoju. Danes je trg na tem področju že popolnoma zasičen, ponudniki pa se v boju s konkurenco trudijo z vedno novimi storitvami. Ker tovrstne storitve zahtevajo izjemno podporo informacijske tehnologije ter čim krajši čas razvoja, je konkurenčnost telekomunikacijskih operaterjev močno odvisna od njihove aplikacijske podpore. Obstoječi sistemi je navadno niso sposobni zagotoviti, zato se pojavljajo potrebe po prenovi. Izvedba obsežnih sprememb v aplikacijski arhitekturi je izredno težka, velikokrat pa se ob tem pozablja, kakšen vpliv imajo te spremembe na preostanek organizacije. Zaradi tega se je tovrstnih projektov bolje lotiti v obliki prenove celotne poslovno-informacijske arhitekture. Ta zajema ne le aplikacijsko arhitekturo, temveč tudi poslovno in tehnološko arhitekturo in je kot taka izjemno široka in zahtevna praksa. Za lažje izvajanje le-te si lahko pomagamo z vrsto orodij in ogrodij. Ker pa je to relativno nova panoga, nobeno izmed njih ne nudi popolne podpore, s katero bi lahko pokrili vse potrebne aktivnosti. Zato smo uporabili kombinacijo ogrodij, s katerimi lahko zadostimo vsem potrebam izvajanja poslovno-informacijske arhitekture tekom njenega celotnega življenjskega cikla. TOGAF smo uporabili za metodologijo. Z njim si torej pomagamo pri načinu samega izvajanja prakse poslovno-informacijske arhitekture. ArchiMate smo uporabili za jezik. Z njim modeliramo arhitekturo in predstavljamo izdelke, ki jih metodologija zahteva. Frameworx uporabimo za vsebino. Z njim si pomagamo pri obvladovanju obsega, ki ga modeliramo. Skupaj tvorijo celoto, s katero lahko bistveno olajšamo izvajanje poslovno-informacijske arhitekture kot prakse ter omogočimo kvalitetno izvedbo prenove. Končni rezultat je nova, učinkovitejša arhitektura, ki organizaciji omogoča večjo konkurenčnost na trgu in je tudi ključna za pripravljenost na bodoče spremembe.

## Ključne besede

*Poslovno-informacijska arhitektura, telekomunikacije, sistemi za podporo poslovanju, TOGAF, ArchiMate, Frameworx*



# Abstract

Until recently, telecommunications were a flourishing industry in rapid development. Today, the market in this industry is completely saturated and providers are dealing with the competition by introducing new services. Since these services require exceptional information technology support and the shortest possible development time, the competitiveness of the telecommunications operators depend heavily on their application support. Legacy systems are usually unable to provide that kind of support, therefore a need for renovation arises. Large-scale changes in the application architecture are demanding, but often the impact they have on the rest of the organization is forgotten. For this reason, such projects are better addressed in the form of enterprise architecture. This includes not only the application architecture, but also business and technology architectures and as such is an extremely broad and challenging practice. To facilitate its implementation a variety of tools and frameworks can be used. However, since this is a relatively new field, none of them provides the full scope, which would cover all the necessary activities. Therefore, we used a combination of frameworks, which can meet all the needs of the implementation of enterprise architecture throughout its entire lifecycle. We used TOGAF for its methodology. It helps us by defining the actual implementation steps. We used ArchiMate for its language. It is needed for architecture modeling and the representation of the methodology-defined output. We used Frameworkx for the content. It helps us to manage the extent of the content that needs to be modeled. Together they significantly facilitate the practice of enterprise architecture and allow for a high quality renovation. The result is a new, more efficient architecture that allows the organization to be more competitive on the market but also provides key readiness for future changes.

## Keywords

*Enterprise architecture, telecommunications, business support systems, TOGAF, ArchiMate, Frameworkx*



# 1 Uvod

Postavitev nove arhitekture je v vsaki večji organizaciji izziv. Je pa to še toliko večji zalogaj tam, kjer je odvisnost od informacijske tehnologije velika in kjer ima vsakršen izpad velik vpliv na poslovanje. Telekomunikacijska podjetja vsekakor spadajo v to kategorijo.

V zadnjih tridesetih letih se je telekomunikacijska industrija korenito spremenila. Podjetja v tej panogi so dala skozi vse, od izjemne rasti do zasičenosti trga. Tekom vseh transformacij v teh letih se je posledično pojavila potreba po večjih spremembah znotraj organizacij, tako poslovnih kot informacijskih. Najprej so bile potrebne hitre širitve in avtomatizacija čim večjega števila procesov, nato pa optimizacija, prilagajanje trgu, razvijanje novih in novih produktov ter razvoj v smer agilnosti.

Zaradi zahtevnosti arhitekturnih projektov so le-ti v telekomunikacijah redko obsežni in tudi redko uspešni. Navadno se zamenjuje le posamezne sisteme ali pa se projekti zavlačujejo in presegajo zadane finančne okvirje. Vsakršna večja dodana vrednost prenove pa zahteva obširne spremembe z jasno vizijo ter kvalitetno izpeljavo del, ki se drži predvidenega proračuna in s tem ne ogroža preostalih planov organizacije.

Za vsako organizacijo je izvedba tovrstnega projekta lažja, če je v njej že vpeljan pojem poslovno-informacijske arhitekture. Poznavanje procesov, vlog, odgovornih, aplikacij in sistemov, na katerih te tečejo, močno olajša tako načrtovanje nove arhitekture kot prehod nanjo. Predvsem pa je na vsak tak projekt potrebno gledati celostno, torej tako s poslovnega kot tehničnega vidika. Vse prevečkrat se spremembe dogajajo zaradi tehničnih razlogov, brez pravega pregleda na vpliv, ki bi ga imele na procese, dogaja pa se seveda tudi obratno, organizacija in njeni procesi se spreminjajo, ne da bi vzeli v obzir obstoječo informacijsko tehnologijo ter njeno sposobnost prilagajanja tem spremembam. Le celostni pregled je lahko podlaga za uspešne arhitekturne spremembe.

Pri tem celostnem pregledu in sistematskem obravnavanju poslovno-informacijske arhitekture si lahko pomagamo z enim izmed temu namenjenih ogrodij. Za potrebe tega dela bomo uporabili TOGAF, eno najbolj razširjenih tovrstnih ogrodij, ki vsebuje metodologijo za

upravljanje s celotnim življenjskim ciklom arhitekture. Vsebuje pa metodologija med drugim tudi vrsto korakov, ki predvidevajo modeliranje določenih aspektov arhitekture in TOGAF sam ne specificira načina, kako modele pripraviti in predstaviti. Pri tem je odlična dopolnitev ArchiMate, ki je bil specifično ustvarjen natanko za to opravilo. Skupaj tvorita usklajen par, ki igra ključno vlogo pri izvajanju aktivnosti s področja poslovno-informacijske arhitekture.

Pri zadani prenovi sistemov za podporo poslovanju v telekomunikacijskem podjetju sta torej TOGAF in ArchiMate idealna izbira za pomoč pri izvedbi celotnega življenjskega cikla arhitekture, od zastavljanja projekta, do vzpostavitve ciljev in vrednot, pa nato do načrtovanja ciljne arhitekture in izvedbe migracije.

Tekom ene iteracije življenjskega cikla poslovno-informacijske arhitekture bomo poskušali prikazati primeren način za izvedbo večjega arhitekturnega projekta. Cilj pa je nova, tako procesna kot tehnična postavitev, ki bo kos zahtevnim tržnim razmeram in čim bolj dolgoročno prilagodljiva.



## 2 Sistemi za podporo poslovanju v telekomunikacijah

V začetku razvoja telekomunikacij je večina podpornih procesov potekala ročno. Sprejem naročil, vzdrževanje inventarja omrežja, izdaja računov in prejem plačil, izvedba naročil, upravljanje z napakami in tudi sama konfiguracija omrežja so bili popolnoma ročni procesi. Z razvojem informacijske tehnologije pa se je pričela njihova avtomatizacija in tako se je najprej razvil OSS.

OSS je kratica za Operational Support Systems ali Operations Support Systems. OSS so t.i. mrežni sistemi in skrbijo za telekomunikacijsko omrežje in storitve. Ta sklop tipično vsebuje sisteme, ki podpirajo aktivnosti zaledne pisarne, ki upravljajo z omrežjem in njegovim inventarjem, skrbijo za zagotavljanje storitev ter za aktivacijo storitev, za konfiguracijo komponent omrežja, upravljanje z napakami ipd.. Uporabljajo jih pretežno podporne skupine v IT oddelkih in skrbniki omrežja ter skupine za storitve jedrnega omrežja.

BSS je novejša kratica in označuje Business Support Systems. BSS sklop vsebuje sisteme, ki podpirajo k uporabniku usmerjene aktivnosti: zajem in oddajo naročil, prodajo in trženje, procesiranje računov, zbiranje plačil, podporo agentom klicnega centra, ipd. Seznam je tu bolj konkreten, ker ni odvisen od ponujenih storitev: sistem za obračunavanje storitev, sistem za upravljanje z naročili, sistem za upravljanje odnosov s strankami ter sistem za upravljanje klicnega centra so BSS aplikacije. Na ta seznam spada tudi centralni produkti katalog, ki pa se ga ne omenja vedno eksplicitno, saj je velikokrat del enega izmed ostalih sistemov. Poleg tega v BSS sklop spada tudi sistem za prijavo in obravnavo napak. Ti sistemi sicer podpirajo aktivnosti zaledne pisarne, vendar so sami procesi proženi s strani strank, zato jih navadno uvrščamo v BSS sklop.

Vendar meja med obema sklopoma vsekakor ni natančno načrtana. V začetkih razcveta telekomunikacij je bila meja jasna. BSS je zajel naročilo, pripravil vse podatke za obračun ter predal naročilo na OSS stran za izpolnitev naročila. Z naglim razvojem storitev ter porastom

obsega poslovanja pa se je pojavila množica funkcionalnosti, ki se razprostirajo iz enega sklopa v drugega.

Vendar sta obe skupini vse prej kot nasprotni, dejansko sta zelo povezani in dandanes vedno bolj težita k sodelovanju, zato se pogosto uporablja kratica OSS/BSS. Ta označuje celoten portfelj informacijskih sistemov potrebnih za podporo telekomunikacijskim storitvam. Nekatere izmed njih ni preprosto uvrstiti, saj ležijo nekje na meji med obema skupinama in skrbijo za prenos procesov iz ene strani na drugo.

Danes so omrežja in storitve veliko bolj napredni in hkrati fleksibilni, produkti, ki jih telekomunikacijska podjetja ponujajo pa veliko bolj raznoliki. OSS in BSS morata zato sodelovati pri vprašanjih kot so “Kaj lahko stranka naroči, glede na svoje obstoječe storitve? Kaj glede na omrežje in povezljivost? Kaj glede na druge razpoložljive vire?” ipd. Ponudba določene storitve je torej stvar pogajanja med komercialnimi produkti, s katerimi upravlja BSS in sposobnostjo OSS-a, da jih izvede.

Sistemi, ki so razpeti med OSS in BSS:

- Zagotavljanje storitev je integrirano skozi celoten OSS/BSS portfelj ter sledi delovanju storitev. S tem pomaga pri zagotavljanju kakovosti storitve, kakršna je predpisan v sporazumu o ravni storitve.
- Storitveni oz. produktni katalogi so prav tako mejni sistemi. Lahko jih ločujemo na komercialne in tehnične kataloge in v tem primeru vsak izmed tipov spada v svoj sklop. Vendar veliko implementacij oba tipa združuje v isto aplikacijo ali pa sta ločena in tesno sklopljena. V tem primeru lahko storitveni oz. produktni katalog prav tako kategoriziramo bodisi v OSS ali BSS ali oboje.
- Upravljanje storitev je orodje za lažji pregled OSS in BSS procesov v primeru bolj zahtevnih storitev. Če je njihova aktivacija in zagotavljanje kompleksno ter zahteva veliko različnih tehničnih resursov, je upravljanje storitev odgovorno za orkestracijo njihove izpolnitve in sprotno obveščanje o status korakov.

Ločevanje tako danes postaja bolj stvar zgodovinskega ali pa organizacijskega pomena. Tradicionalno so namreč telekomunikacijska podjetja vedno imela informacijske in omrežne sektorje, ki so si delili skrbništvo nad celotnim tehničnim portfeljem.

Kljub današnjem usklajenem delovanju bi bila prenova celotnega portfelja vseeno prevelik zalogaj. Tako za potrebe te naloge ostajamo pri prenovi poslovnega dela ter tistih najnujnejših sistemov, na katere ima prenova največji vpliv.

V okviru tega dela se kot BSS sklop jemlje sistem za obračunavanje storitev, sistem za upravljanje z naročili, sistem za upravljanje odnosov s strankami ter centralni produktni katalog. Naj še poudarim, da se tu pričakuje, da sistem za upravljanje odnosov s strankami vsebuje vse potrebne funkcionalnosti za podporo aktivnostim klicnega centra.

Sistemi za podporo poslovanju je izraz pod katerim je združena vsa programska oprema, ki se jo v telekomunikacijskih podjetjih uporablja za podporo uporabnikom. To je programska oprema za obračunavanje in zaračunavanje, upravljanje s strankami, avtomatizacijo klicnega centra, načrtovanje in upravljanje s produkti, prodajo in trženje ter naročila in njihovo aktivacijo. Sem lahko štejemo tudi aplikacije, ki načeloma že spadajo v OSS del, če so aktivnosti, ki jih podpirajo, sprožene direktno preko stika s stranko.

BSS sklop podpira štiri ključne skupine procesov:

- Upravljanje s prihodki, ki vsebuje obračunavanje, zaračunavanje, poravnavo, mediacijo, zagotavljanje prihodkov, plačila in upravljanje s prevarami.
- Upravljanje s strankami, ki vsebuje skrb za stranke, upravljanje z uporabniško izkušnjo, aktivnosti klicnega centra ter upravljanje odnosov s poslovnimi partnerji.
- Upravljanje s produkti, ki vsebuje razvoj, prodajo in upravljanje s produkti, in ponudbami. Vključuje ustvarjanje storitev, platform za razvoj storitev, produktne kataloge in rešitve za trženje.
- Upravljanje z naročili, ki vsebuje sprejem in ravnanje z naročili. Pogosto je del upravljanja s strankami.



### 3 Poslovno-informacijska arhitektura

Kaj sploh je poslovno-informacijska arhitektura?

Po definiciji iz ISO 15704 standarda je arhitektura opis osnovne razporeditve in povezav med deli sistema (fizičnega ali konceptualnega objekta oz. entitete) [5].

Navadno ima arhitektura več pomenov v odvisnosti od konteksta [11]:

- Formalen opis sistema na ravni komponent, ki vodi implementacijo
- Struktura komponent, razmerij med njimi ter principi in vodila za upravljanje z njihovim oblikovanjem in razvojem
- Organizacijska struktura sistema ali komponente.

Lankhorst jo na primer opredeljuje kot fundamentalno organizacijo sistema, vključenega v svoje komponente, razmerja med njimi in okolje ter principe, ki vodijo njegov načrt in evolucijo.

Iz tega izhaja tudi njegova definicija poslovno–informacijske arhitekture: je koherenten nabor principov, metod in modelov, ki se uporabljajo pri načrtovanju in realizaciji organizacijske strukture, poslovnih procesov, informacijskih sistemov in infrastrukture organizacije [9].

Kot taka poslovno-informacijska arhitektura predstavlja temelje načrtovanja poslovnih sistemov in se je izkazala za ključno orodje, ki deležnikom pomaga pri upravljanju systemskega razvoja in sprememb. Ne pokriva le informacijske tehnologije temveč je predvsem strateško organizacijski izziv [2].

Gartner v svojem slovarju informacijske tehnologije poslovno-informacijsko arhitekturo opredeljuje kot disciplino za proaktivno in holistično vodenje odzivov organizacije na moteče sile z identifikacijo in analizo izvajanja spremembe v smeri željene poslovne vizije in rezultatov. Po tej definiciji poslovno-informacijska arhitektura prinaša dodano vrednost s tem, ko vodjem v poslovnih in tehnoloških oddelkih predstavlja predloge za prilagoditve usmeritev in projektov za doseganje začrtanih ciljev, ki izkoriščajo tovrstne poslovne motnje. Poslovno-

informacijsko arhitekturo se tako uporablja za usmerjanje odločitev v smer proti evoluciji k bodočemu stanju arhitekture [16].

Poslovno-informacijska arhitektura je organizacijska logika, ki odraža integracijske in standardizacijske zahteve podjetja, in ki ji sledijo aplikacije, podatki in tehnična infrastruktura. Le-te so predstavljene v obliki usmeritev in tehničnih odločitev, katerih namen je izvajanje poslovne strategije podjetja [12].

Taka definicija je res izjemno široka, vendar če jo apliciramo na Lankhorstovo definicijo, se obe opredelitvi zelo lepo ujemata. S tem, ko imamo jasno opredeljene gradnike, način gradnje in uporabe, so spremembe in odzivi enostavnejši, prav tako pa tudi zastavljanje novih ciljev in pot do njih. Vse naštetu je predmet poslovno-informacijske arhitekture.

Kljub razmeroma skladnim definicijam pa še vedno velja, da je poslovno-informacijska arhitektura pojem, ki ljudi bega in za katerega imamo veliko različnih interpretacij. To je najbrž posledica tega, da gre za zelo mlado disciplino. Posledično je razvoj šele dodobra začel, poskusov in rezultatov pa veliko. Razumevanje in sprejemanje ter uporaba sta zato precej omejena.

Tradicionalno prakso poslovno-informacijske arhitekture povezujemo z IT oddelki. Ljudje, ki jo primarno izvajajo navadno organizacijsko spadajo vanje in zato je še danes viden precejšen poudarek na informacijsko—tehnološkem delu celotne poslovno-informacijske arhitekture.

Kljub tehničnem poudarku lahko z organizacijskega vidika identificiramo več deležnikov znotraj ter izven podjetja, od najvišjih slojev menedžmenta do programskih inženirjev. Vsak deležnik potrebuje specifične informacije predstavljene v dostopni obliki, da se lahko prilagodi vplivu tako široko obsegajočega dogajanja. Pridobiti celosten pregled takih sprememb ter njihovih vplivov je nujno vendar izjemno zahtevno [8].

In tudi sicer je praksa poslovno-informacijske arhitekture zelo zahtevna praksa. Potreba po poslovno-informacijski arhitekturi se namreč navadno pokaže predvsem pri večjih organizacijah, ki imajo veliko število procesov, vlog in sistemov, ki pa so le redko dokumentirani. V teh primerih je potreben obsežen začetni popis, kar pa zahteva velik delovni vložek. Vendar tudi sprotno vzdrževanje ažurnosti podatkov ni zanemarljivo opravilo.

Se pa navadno ves trud izkaže za poplačanega, saj so prednosti uporabe poslovno-informacijske arhitekture številne:

- Nudi podporo ob organizacijskih spremembah, na primer ob prevzemih ali združevanju z drugimi organizacijami

- Sili k standardizaciji poslovnih procesov in omogoča konsolidacijo, ponovno uporabo in integracijo procesov
- Pri upravljanju s projektnim portfeljem pomaga pri odločanju in prioritizaciji
- Izboljšuje kolaboracijo in komunikacijo med deležniki projektov ter prispeva k boljši opredelitvi obsega in željenih projektnih rezultatov
- Pohitri zbiranje zahtev in izboljšuje njihovo kvaliteto
- Med sistemskim razvojem in testiranjem pripomore k bolj optimalnim sistemskim načrtom in učinkoviti razporeditvi resursov
- Pomaga pri uveljavitvi discipline in standardizacije na področju aktivnosti planiranja IT-ja ter k hitrejšim odločitvam na tem področju
- Pomaga pri zmanjševanju razvojnih in operativnih stroškov ter minimizira podvajanje storitev
- Zmanjšuje kompleksnost informacijske tehnologije ter pripomore h konsolidaciji aplikacij in podatkov ter k boljšemu sodelovanju med sistemi
- IT naredi bolj odprt in odziven, izboljša dostopnost podatkov ter večja transparentnost pri infrastrukturnih spremembah
- Zmanjšuje poslovno tveganje zaradi sistemskih napak in kršitev varnosti, prav tako znižuje tveganje ob izvedbi projektov [19]

Na kratko, poslovno-informacijska arhitektura predstavlja temelje, na katerih lahko zgradimo poslovne procese in tehnološko podporo zanje, ki bodo omogočali dandanes podjetjem tako potrebno agilnost.

Načrtovanje velikih sistemov zahteva t.i. »top-down« pristop, kjer se začne z visokim nivojem abstrakcije in nadaljuje s podrobnostmi na nižjih nivojih. Tako se zagotovi celosten pregled in konsistenca. Za organizacijo podsistemov in modulov je potreben predpisan mehanizem, ki se nato uporablja tako za popis trenutnega stanja kot za želeno novo stanje in tudi migracijsko pot, preko katere iz trenutnega preidemo na želeno stanje.

Na tem mestu se dotaknimo še modeliranja poslovno-informacijske arhitekture. Da bi bil popis resnično uporaben, mora biti berljiv, to pa lahko dosežemo samo s specifično notacijo. Organizacije tipično delimo na tri sloje: poslovni, aplikativni in tehnološki. Z druge strani pa lahko vsakega izmed njih opredelimo preko informacijskega, vedenjskega in strukturnega vidika. Notacija mora omogočati konsistentno modeliranje arhitekture po vseh slojih in z vseh vidikov ter tudi pogled nanjo z različnih zornih kotov.

Na poslovno-informacijsko arhitekturo lahko gledamo tudi kot na način komunikacije med deležniki. Če je homogeno predstavljena, različnim sektorjem lajša razumevanje med vsebinsko-organizacijskimi prehodi. Ker so vsi vidiki in nivoju predstavljeni s sorodnimi

koncepti, je komunikacija med predstavniki različnih organizacijskih enot ali različnih vlog lažja. Dodatno uporaba takega zapisa lajša začetni visokonivojski popis in predstavlja podlago za kasnejši prehod na zbiranje zahtev na nižjih nivojih. Visokonivojski pogled pa obenem omogoča širši pogled, celo na nivoju celotne arhitekture, kar prinaša možnost ugotavljanja vplivov določenih zahtev na ostala področja organizacije. Naloga arhitekta je v tem primeru dokumentacija prednosti in slabosti takih zahtev in morebitnih prilagoditev za omiljenje slabosti. Brez poslovno-informacijske arhitekture in njenega širokega homogenega pogleda, bi se take posledice zelo verjetno spregledalo, zastopanje interesov vseh delov organizacije pa ne bi bilo enakomerno. Poslovno-informacijska arhitektura je kot taka naravnana zelo v prihodnost. Njen primarni sestavni del je sicer popis trenutnega stanja, vendar je njegov namen podpora spremembam in prilagoditvam. S tem je njena vloga tudi osnova za načrtovanje in analizo bodočega stanja.

Dodatno lahko preko poslovno-informacijske arhitekture ocenimo zrelost organizacije. Razdelimo jih lahko v štiri nivoje razvitosti in nadzora [13]:

- Poslovni silos – podjetja poskušajo neodvisno maksimirati zadovoljitev potreb vsake posamezne funkcijske ali poslovne enote, kar povzroča delitve in pomanjkanje integracije med različnimi iniciativami znotraj podjetja.
- Standardizirana tehnologija – povečuje učinkovitost skozi standardizacijo in centralizacijo upravljanja s tehnologijo
- Optimizirano jedro – podatki in procesi so standardizirani za podjetje kot celoto.
- Poslovna modularnost – podjetja ponovno uporabljajo obstoječe komponente glede na potrebe poslovnih procesov s ciljem ohranitve globalnih standardov in lokalnih raznolikosti.

Te kategorije pa niso postavljene glede na neko specifično karakteristiko, ampak označujejo organizacije kot celote, torej njihove poslovno-informacijske arhitekture.

Podjetja torej napredujejo iz enega nivoja razvitosti v drugega s spremembo v vzorcih investicij v IT ter z implementacijo novih praks za upravljanje z arhitekturo, s formalizacijo učenja organizacije o izkoriščanju kapacitet IT in uvajanjem sprememb v poslovnih procesih. Navadno so bolj razvita podjetja bolj uspešna pri doseganju strateških ciljev in imajo višje donose na vložen kapital [13].

Kot rečeno je celostno izvajanje poslovno-informacijske arhitekture izjemno široko, kompleksno in zahtevno. V pomoč pri uporabi te prakse so zato nastala številna ogrodja. Arhitekturno ogrodje je vrsta pripomočkov za popis informacijskih sistemov z določenimi



gradniki, prikazi uporabe gradnikov, poenoteno terminologijo in nasploh vsem, kar lahko organizaciji pomaga pri izvajanju življenjskega cikla arhitekture.

V osemdesetih se je v Združenih državah Amerike in v Evropi razcvetel razvoj ogrodij za poslovno-informacijsko arhitekturo. Izmed njih so najbolj znani Computer Integrated Manufacturing Open System Architecture (CIMOSA), od koder tudi izvira izraz poslovno-informacijska arhitektura, Purdue Enterprise-Reference Architecture (PERA), GIM arhitektura in ARIS. Zachmanovo ogrodje prav tako spada v to skupino in je bilo prvo širše uporabljano ogrodje, ki ga še danes pogosto jemljemo kot referenco. Strukturirano predstavlja različne koncepte tako, da so prilagojeni zornim kotom različnih deležnikov podjetja.

Danes pa je eno izmed najbolj priljubljenih in celoviti ogrodij TOGAF, ki se razvija pod okriljem organizacije The Open Group.



## 4 TOGAF 9.1

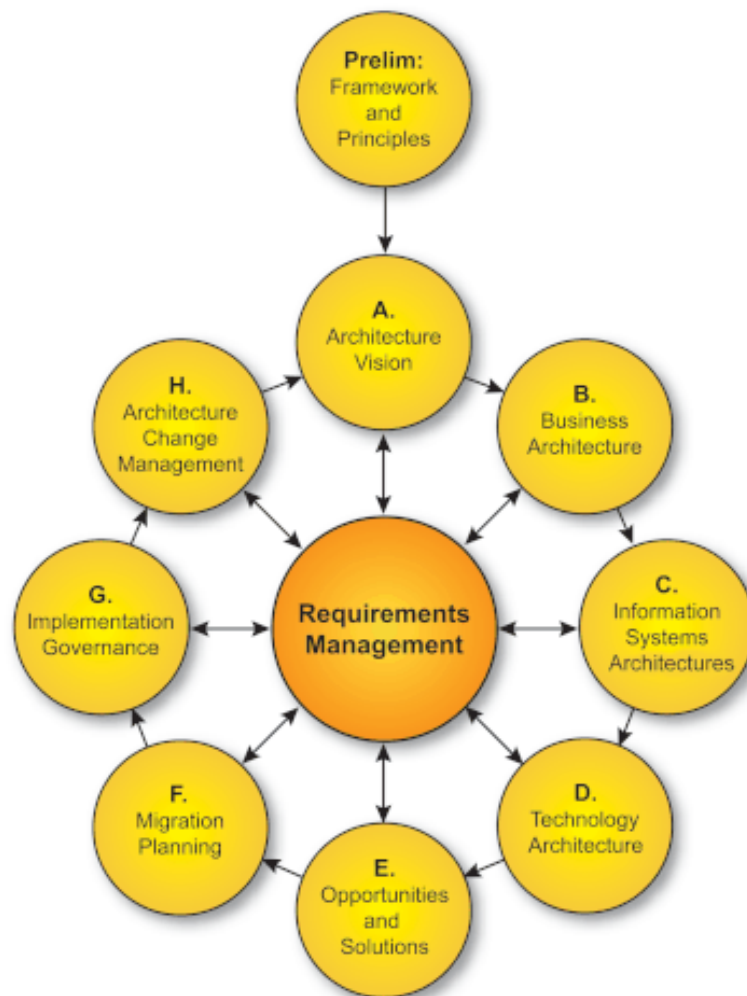
### 4.1 Uvod v TOGAF

The Open Group Architecture Framework ali skrajšano TOGAF, je ogrodje za poslovno-informacijsko arhitekturo, ki vključuje pristop za načrtovanje, planiranje, implementacijo in upravljanje poslovno informacijske arhitekture.

TOGAF se je začel razvijati na začetku devetdesetih kot metodologija za razvoj tehnične arhitekture. Prva verzija je izšla leta 1995. Nato se je pod okriljem The Open Group razvil v visokonivojsko ogrodje, ki pokriva celotno poslovno-informacijsko arhitekturo. Prva verzija takega obsega je bila verzija 8 leta 2002. Trenutno zadnja, verzija 9.1, je izšla leta 2001 [18].

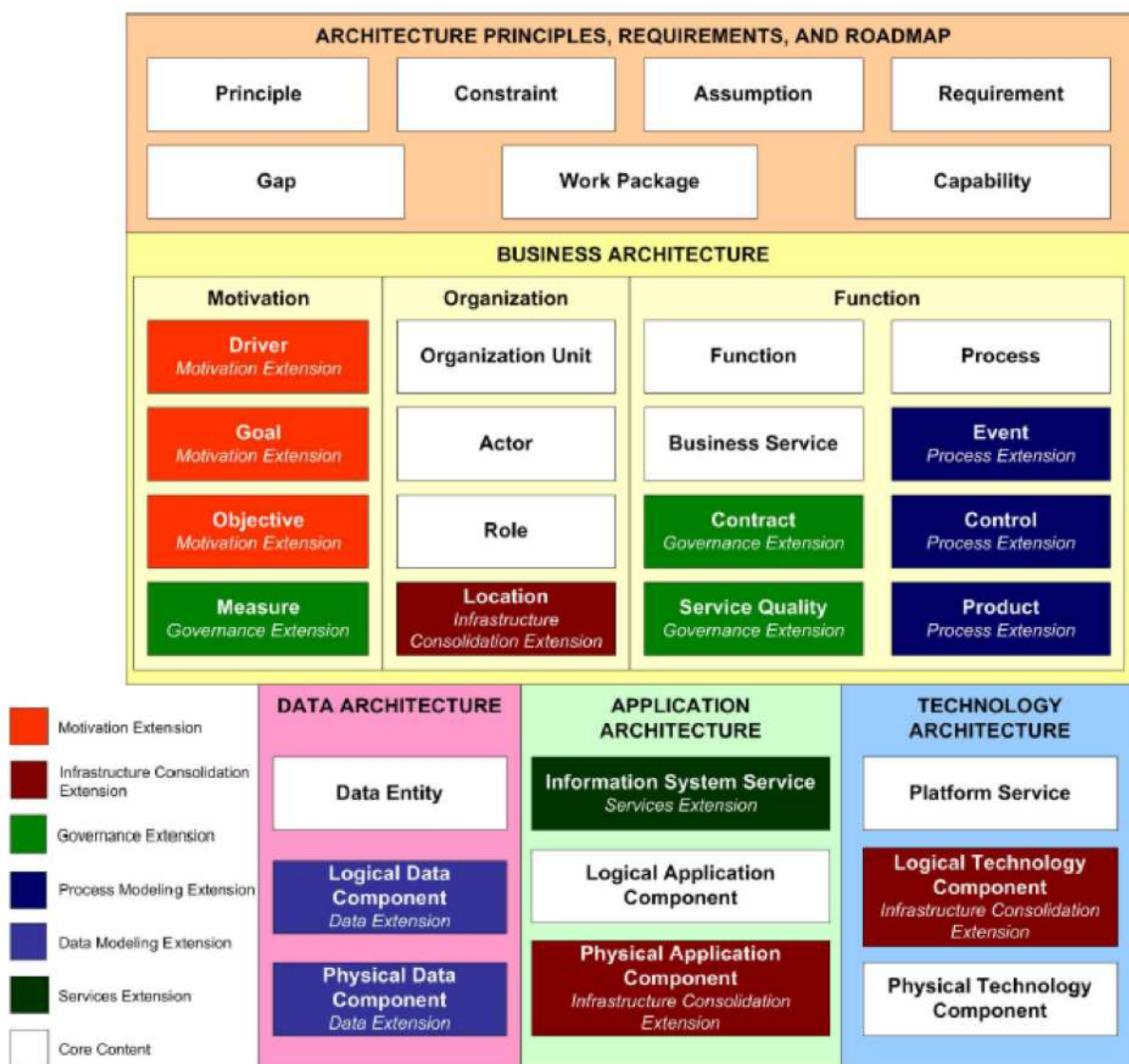
Jedro TOGAF ogrodja predstavlja Architecture Development Method, metodologija za razvoj arhitekture ali krajše ADM. Ta pokriva vse potrebne korake za vzpostavitev arhitekture na štirih temeljnih TOGAF domenah: poslovni, aplikacijski, podatkovni in tehnološki.

Poleg preliminarne faze, ki je nekakšen uvod v poslovno-informacijsko arhitekturo, ADM vsebuje še osem faz, ki pokrivajo tako zbiranje zahtev kot načrtovanje vseh štirih domen, končno implementacijo in upravljanje nove arhitekture. Teh osem faz pokriva celotni življenjski cikel poslovno-informacijske arhitekture, ki se ciklično ponavlja ter s tem podpira konstantno iterativno izboljševanje arhitekture (slika 4-1: Cikel razvoja arhitekture po TOGAF-u [11]).



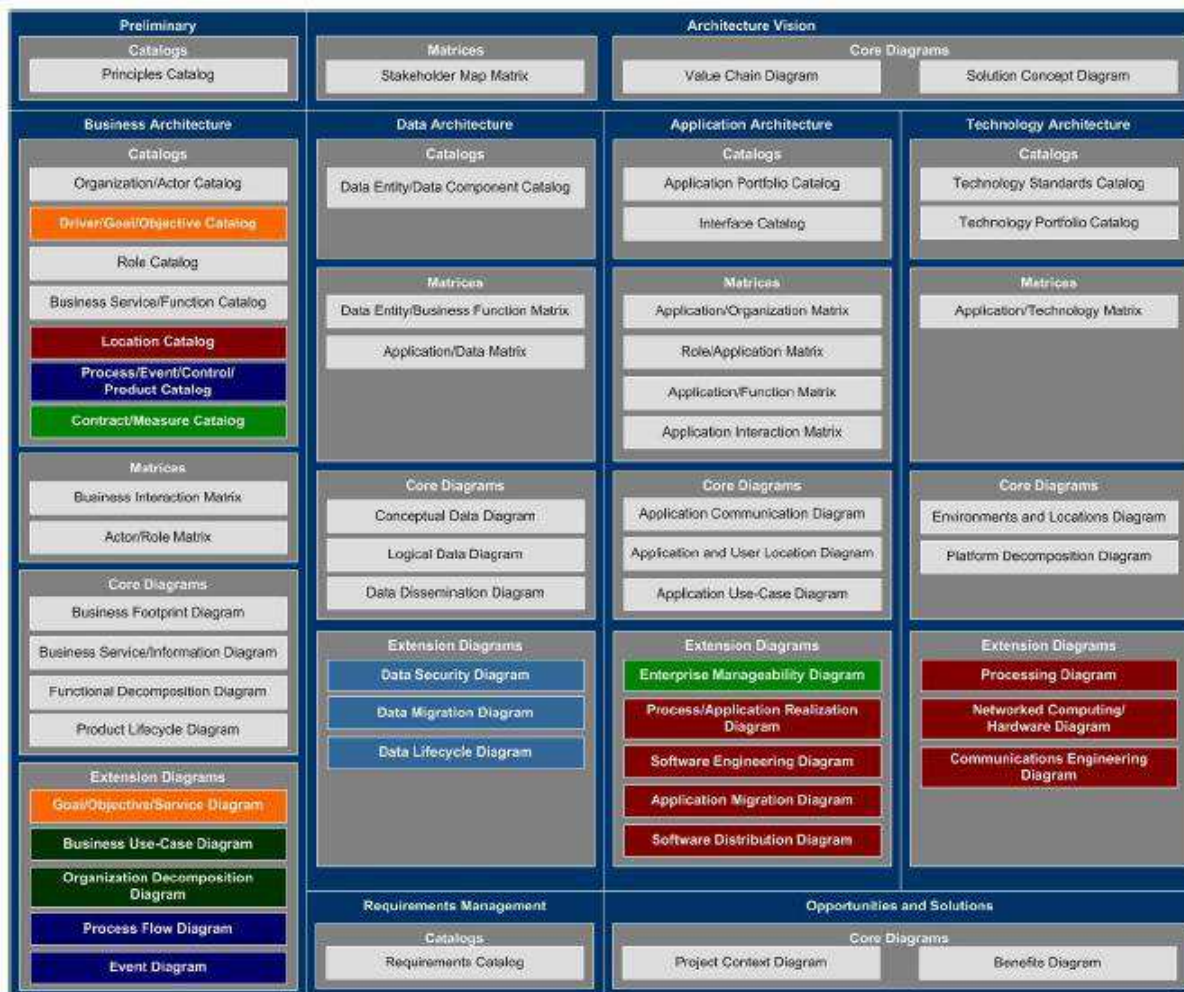
*Slika 4-1: Cikel razvoja arhitekture po TOGAF-u [11]*

Med izvajanjem metodologije le-ta zahteva vrsto izdelkov, ki naj bi bili rezultat posameznih faz in vhod v naslednje faze. Ti izdelki so oz. vsebujejo tudi procesne tokove, plane, zahteve ipd. Zato je del TOGAF-a tudi vsebinsko ogrodje, ki vsebuje model za arhitekturno vsebino. Njegov namen je, da so izdelki, ki nastajajo tekom metodologije, konsistentni in metodično strukturirani. Del tega ogrodja je tudi meta model, ki dodatno skrbi za strukturiranost (slika 4-2: Entitete TOGAF vsebinskega meta modela [11]).



Slika 4-2: Entitete TOGAF vsebinskega meta modela [11]

Vsebinski meta model vsebuje vse elemente, ki jih lahko uporabimo za opredelitev poslovno-informacijske arhitekture, vključno z relacijami med njimi. S tem predstavlja strukturo, ki jo lahko uporabimo za pripravo vhodov in izhodov vsake faze.



Slika 4-3: Povezava med artefakti in meta modelom[11]

Podobno lahko uvrstimo artefakte oz. izdelke, ki jih pripravljamo tekom izvajanja metodologije in so po TOGAF-u specificirani za vsak korak ADM-a. Veliko izmed izdelkov prikazanih na sliki 4-2: Entitete TOGAF vsebinskega meta modela [11], bomo podrobneje spoznali v poglavjih 8 do 13.

## 4.2 Preliminarna faza

Preliminarna faza pokriva aktivnosti, ki vpletene pripravijo na novo arhitekturo. Določiti je potrebno, kakšne so želje, kakšne sposobnosti, na kakšnih principih želimo zgraditi novo arhitekturo, kakšen je namen spremembe, kakšne obstoječe ali nove standarde želimo uporabiti pri tem ter seveda v kakšnem obsegu naj bo sprememba. Prav tako se določi ekipa, ki bo delala na celotnem projektu.

Po opravljeni preliminarni fazi naj bi tako imeli organizacijski model za poslovno-informacijsko arhitekturo, prilagojeno ogrodje, začetno verzijo arhitekturnega repozitorija, izjavo o poslovnih principih, ciljih in gonilu, ki stoji za odločitvijo za tak projekt.

### 4.3 Faza A: arhitekturna vizija

Namen faze A je priprava vizije in poslovne vrednosti, ki naj bi jo prinesla nova arhitektura. V tej fazi se tudi pridobiva odobritev za okvirni program dela.

V tej fazi se projekt priprave poslovno-informacijske arhitekture tudi uradno začne. Najprej je potrebno identificirati deležnike, zahteve in pomisleke, nato pa potrditi ter dodelati principe, cilje in gonila iz preliminarne faze. V sklopu te faze se potrdi izjava o delu, oceni zmožnosti, dodela vizijo, pripravi osnutek arhitekturne definicije in definira plan, kako naj bi se komuniciralo znotraj projekta.

### 4.4 Faze B, C in D: poslovna, informacijska in tehnološka arhitektura

V fazah B, C in D definiramo želene arhitekture na vseh štirih domenah. V fazi B definiramo poslovno arhitekturo, v fazi C aplikacijsko in podatkovno arhitekturo, čemur skupaj pravimo informacijska arhitektura, in nazadnje v fazi D še tehnološko arhitekturo.

V sklopu tega dela se bomo osredotočili na te tri faze ob predpostavki uspešno zaključene preliminarne faze in faze A.

#### 4.4.1 Faza B: poslovna arhitektura

Poslovna arhitektura opredeljuje način, kako deluje organizacija.

V sklopu faze B je potrebno opisati trenutno poslovno arhitekturo ter pripraviti modele poslovnih procesov, primere uporabe in razredne modele, na podlagi katerih se sestavi ciljna poslovna arhitektura. Nato se izvede analiza vrzeli in pripravi seznam aktivnosti, ki jih bo potrebno izvesti za uspešen prehod iz obstoječe na ciljno arhitekturo.

Rezultat faze B je prva verzija dokumenta arhitekturne definicije, ki lahko med drugim vsebuje podroben opis poslovnih funkcij, informacije, ki jih potrebujejo, pravila in vodila, ki jih organizacija upošteva, delovne prakse, zakonodajo, matriko znanj in opis delovnih mest. Arhitekturna definicija pa med drugim vsebuje tudi poglede na arhitekturo, kakršne imajo vsi vpleteni deležniki.

#### 4.4.2 Faza C: informacijska arhitektura

Kot rečeno faza C pokriva dve domeni in sicer aplikacijsko ter podatkovno arhitekturo. Začnemo lahko bodisi z eno ali drugo. V primeru, da so določene aplikacije identificirane kot kritične, je bolje začeti z aplikacijsko arhitekturo, saj bo le ta vodilo za podatkovno. V nasprotnem primeru je začetek podatkovno arhitekturo primernejši, saj ni pristranski do različnih aplikacij.

Za obe domeni je potrebno pripraviti obstoječo in ciljno arhitekturo, analizo vrzeli ter plan aktivnosti za prehod iz obstoječih v ciljne arhitekture. Prav tako je potrebno identificirati vpliv nove arhitekture na celotno organizacijo.

V sklopu razvoja podatkovne arhitekture je v dokument definicije arhitekture potrebno dodati poslovni podatkovni model, logični podatkovni model, procesni model za upravljanje s podatki in seznam podatkovnih entitet, ki se uporabljajo za posamezno poslovno funkcijo.

V sklopu razvoja aplikacijske arhitekture je v dokument definicije arhitekture potrebno dodati portfelj aplikacij in njihovih vmesnikov, matriko uporabe aplikacij glede na vlogo uporabnika, poslovno funkcijo ter organizacijo, diagram komunikacije med aplikacijami, diagrame uporabe za aplikacije, diagram realizacije procesov ipd.

#### 4.4.3 Faza D: tehnološka arhitektura

Podobno kot pri fazah B in C je tudi tu potrebno definirati obstoječo in ciljno arhitekturo, izdelati analizo vrzeli med njima in pripraviti seznam aktivnosti, ki bodo potrebne za prehod, pri čemer tehnološka arhitektura pokriva tehnološko domeno. Pod tehnološko domeno pojmuje platforme, logične tehnološke komponente, lokacijo komponent, načine povezovanja in varnostne mehanizme.

V sklopu faze D je tako potrebno pripraviti katalog uporabljenih standardov in tehnološki portfelj, matriko uporabljene tehnologije za posamezno aplikacijo, lokacijske diagrame, diagram dekompozicije platform, omrežni diagram in diagram komunikacije.



## 4.5 Faze E do H: rešitve, plan migracije, implementacija, upravljanje s spremembami

### 4.5.1 Faza E: priložnosti in rešitve

V fazi E se išče rešitve, s katerimi lahko pridemo do ciljne arhitekture. V obzir je treba vzeti vse analize vrzeli iz faz B, C in D ter tudi aktivnosti, ki so bile identificirane kot potrebne v teh fazah ter jih grupirati v logične skupine.

### 4.5.2 Faza F: planiranje migracije

Faza E je praktično pripravljalna faza na fazo F. Identificirane logične skupine sprememb v fazi F uporabimo za pripravo natančnega plana migracije. Plan mora vsebovati potrebe po resursih in časovnico aktivnosti.

### 4.5.3 Faza G: upravljanje implementacije

Faza G vsebuje dejanski prehod na novo arhitekturo. Za čim bolj uspešen prehod je potrebno potrditi njegov obseg in prioritete, identificirati potrebna znanja in resurse za vsak korak, voditi namestitvev posamezne rešitve ter na koncu izvesti še pregled ustreznosti, uvesti novo postavitev v izvajanje in zapreti implementacijo.

### 4.5.4 Faza H: upravljanje s spremembami

Namen faze H je vzdrževanje arhitekture. Vsaka postavitev namreč ni končna in prej ali slej zahteva določene spremembe. Spremembe lahko razdelimo v tri kategorije:

- Poenostavitev: ta tip spremembe lahko navadno izvedemo kar v okviru upravljanja s spremembami
- Postopna sprememba: gre za tipične spremembe v okviru življenjskega cikla arhitekture, bodisi zaradi sprememb v poslovanju ali organizaciji. Odvisno od vsebine oz. obsega spremembe jo lahko realiziramo bodisi v okviru upravljanja s spremembami ali z novim ciklom razvoja arhitekture.
- Jedrne spremembe: gre dejansko za novo zastavljeno arhitekturo. Tak tip spremembe zahteva nov cikel razvoja arhitekture.

Za spremembe, ki jih lahko realiziramo v okviru upravljanja s spremembami je postopek sledeč: oceniti je potrebno vrednost spremembe, vpeljati orodja za spremljanje postopka, identificirati

tveganja, izpeljati analizo želene spremembe, pripraviti zahteve ter pripraviti in nato izvesti načrt implementacije.

## 4.6 TOGAF dokument

Kot rečeno ADM predstavlja jedro TOGAF ogrodja, vendar še zdaleč ni vse. TOGAF poleg metodologije vsebuje še smernice in tehnike za izvajanje ADM ter naslednje sklope:

- Ogrodje arhitekturne vsebine (Architecture Content Framework) – vključuje meta model za arhitekturne artefakte, način večkratne uporabe arhitekturnih gradnikov in pregled tipičnih arhitekturnih izsledkov
- Organizacijski kontinuum (Enterprise Continuum) – opredeljuje primerne taksonomije in orodja za kategorizacijo in shranjevanje izdelkov, nastalih v toku aktivnosti priprave arhitekture
- Referenčne modele – vsebuje nabor referenčnih modelov, vključno s TOGAF Foundation Architecture in III-RM (Integrated Information Infrastructure Reference Model).
- Okvir arhitekturnih zmožnosti (Architecture Capability Framework) – obravnava organizacijo, njene procese, vloge, znanja in odgovornosti potrebne za vzpostavitev in delovanje določene arhitekturne funkcije

Z namenom vzpostavitve nove arhitekture se bomo osredotočili na jedro, torej metodologijo samo.

## 5 ArchiMate 2.1

Danes je v uporabi vrsta jezikov za organizacijsko in procesno modeliranje: ebXML, ARIS, Testbed, BPML, BPMN, UML ipd. Večina omenjenih jezikov ponuja koncepte za modeliranje na primer detajlnih poslovnih procesov, ne pa visokonivojskih razmerij med temi procesi, zato niso ravno primerni za modeliranje celotnih arhitektur [4].

ArchiMate je jezik za modeliranje poslovno-informacijske arhitekture. Nastal je pod okriljem nizozemskega Telematica Instituuta, ki je ArchiMate razvijal od leta 2002 do leta 2004. Leta 2008 je lastništvo prevzel The Open Group, ki je nato leto kasneje izdal verzijo 1.0. Zadnja verzija je izšla leta 2013 in sicer verzija 2.1 [17].

Na kratko, ArchiMate omogoča modeliranje produktov, storitev, ki jih ti produkti omogočajo, ter procesov, aplikacij in tehnološke podlage, s katerimi so podprti. Za razliko od drugih modelirnih jezikov, ki so specifično namenjeni modeliranju določene domene ali koncepta, lahko z ArchiMate jezikom opišemo tako poslovne procese, organizacijske strukture, tehnično infrastrukturo kot aplikacije.

Je relativno nov jezik, ki nudi gradnike in koncepte, ki so potrebni za celostno obravnavanje organizacij. Razvit je bil za opisovanje in analizo poslovno-informacijske arhitekture na visokem nivoju. Vsebuje meta model teh gradnikov in konceptov ter grafično notacijo za njihovo predstavitev skozi vse tri sloje arhitekture. Z njim lahko modeliramo tudi razmerja in odvisnosti med posameznimi komponentami.

Vendar ArchiMate ni samo jezik, je celotno modelirno ogrodje, podobno kot TOGAF, ki deli poslovno-informacijsko arhitekturo na tri ravni: poslovno, aplikacijsko in tehnološko.

Poslovna raven strankam ponuja produkte in storitve. Ti produkti in storitve so realizirani v okviru poslovnih procesov, ki jih izvajajo poslovni akterji. Poslovna raven torej opisuje poslovne procese, funkcije, dogodke, storitve in poslovne enote.

Aplikacijska raven podpira poslovno raven z aplikacijskimi storitvami, ki so realizirane v obliki programskih aplikacij.

Tehnološka raven nudi infrastrukturne storitve, ki so potrebne za izvajanje programskih aplikacij. Opisuje torej strojno in komunikacijsko infrastrukturo.

Na vsaki ravni nato upoštevamo tri aspekte, ki predstavljajo naravni jezik: aktivne strukture (osebke oz. subjekte), vedenja (povedke) in pasivne strukture (predmete oz. objekte). Ti aspekti predstavljajo tri glavne tipe elementov, iz katerih je sestavljeno jedro jezika ArchiMate. Elementi aktivne strukture so npr. poslovni akterji, aplikacijske komponente in naprave, ki opravljajo določene naloge oz. aktivnosti. Vedenja so nato aktivnosti, ki jih izvajajo elementi aktivne strukture. Elementi pasivne strukture pa so predmeti, nad katerimi se izvajajo te iste aktivnosti. To so navadno podatkovni elementi, lahko pa tudi fizični objekti [10].

## 5.1 Struktura

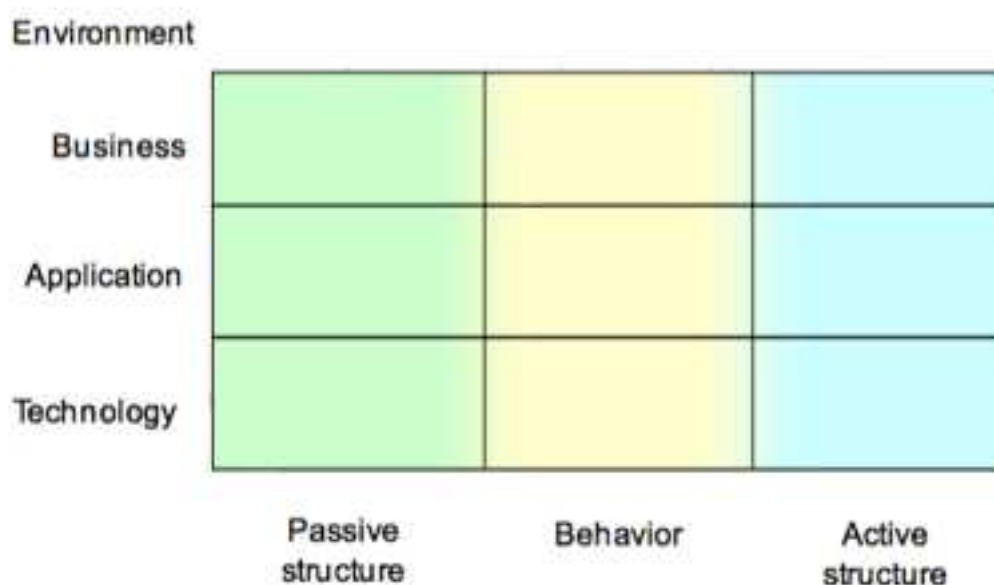
Celotna ArchiMate-ova struktura temelji na njegovem meta modelu. Metamodel sestavlja vrsta tako imenovanih konceptualnih domen, izmed katerih vsaka predstavlja določeno področje. Izbira teh konceptualnih domen temelji na domenah tipičnih arhitekturnih ogrodij kot sta TOGAF in Zachmanovo ogrodje ter na podlagi arhitekturnih praks podjetij, ki so sodelovala v projektu ArchiMate.

Konceptualne domene so naslednje:

- Produktna domena: vsebuje koncept »produkt«, ki opisuje produkte ali storitve, ki jih organizacija nudi svojim strankam
- Organizacijska domena: opisuje poslovne akterje in vloge, ki jih zavzemajo
- Procesna domena: opisuje poslovne procese in funkcije, ki jih sestavljajo poslovne aktivnosti
- Informacijska domena: predstavlja znanje v organizaciji ter to, kako je znanje strukturirano
- Podatkovna domena: znotraj nje so informacije strukturirane tako, da so primerne za avtomatsko procesiranje
- Aplikacijska domena: opisuje programske aplikacije, ki s svojimi storitvami podpirajo poslovanje
- Domena tehnične infrastrukture: vključuje koncepte za platforme strojne in komunikacijske opreme, ki jih aplikacije potrebujejo za svoje delovanje [7].

Vsako konceptualno domeno lahko z določeno mero generalizacije uvrstimo v eno izmed treh ArchiMate ravni.

Poslovna raven opisuje poslovne elemente ter razmerja med njimi znotraj domene poslovne arhitekture. Znotraj te domene so ključni elementi sledeči: namen, produkt, poslovni proces/funkcija/interakcija, akter/vloga, poslovna storitev, lokacija ipd. Ti elementi lahko pokrijejo produktno, organizacijsko, procesno in informacijsko domeno. Aplikacijska raven opisuje arhitekturo aplikacij, torej programske opreme, ter razmerij med njimi. Znotraj te domene so ključni elementi sledeči: podatkovni objekt, aplikacijska storitev, aplikacijska komponenta, aplikacijski vmesnik ipd.. Ti elementi lahko pokrijejo Podatkovno in aplikacijsko domeno. Vloga aplikacijske ravni je podporna in sicer omogoča izvajanje poslovne ravni. Tehnološka raven opisuje tehnološko arhitekturo oz. infrastrukturo, na kateri se izvaja aplikacijska raven. Govorimo torej o strojni opremi ter razmerji med različnimi komponentami strojne opreme. Znotraj te domene so ključni elementi sledeči: artefakt, infrastrukturna storitev, vozlišče, mrežna/komunikacijska pot ipd. Ti elementi lahko pokrijejo domeno tehnične infrastrukture. Vloga tehnološke ravni je prav tako podporna in sicer omogoča izvajanje aplikacijske ravni s tem pa tudi posredno poslovne ravni.



Slika 5-1: Jedro ArchiMate ogrodja [10]

Aspekti in ravni skupaj tvorijo ArchiMate ogrodje v obliki devetih celic. Namen ArchiMate-a je nuditi modelirni jezik, ki bi bil sposoben primerno predstaviti kompleksnost vseh arhitekturnih domen in relacij med njimi. Za to predstavitev uporablja poglede in vidike skladno

z definicijo le-teh po IEEE standardu 1471. Predstavljena struktura tako omogoča modeliranje iz različnih zornih kotov in predvsem predstavitev iz različnih zornih kotov oz. t.i. vidikov (viewpoints). Vidik je pozicija, ki jo definirata obe dimenziji – raven in aspekt. Kot tak predstavlja zorni kot določenega deležnika oz. del njegovega želenega pogleda. Tipično se deležniki namreč zanimajo za enega ali več vidikov, vendar vsekakor na manjšo količino. Praktično nobenega deležnika ne zanimajo vsi pogledi poslovno-informacijske arhitekture.

ArchiMate je zato primeren za visokonivojsko modeliranje poslovnih procesov znotraj poslovnega dela arhitekture. Za bolj natančno modeliranje poslovnih procesov pa je še vedno bolj primeren kakšen jezik, ki je namenjen specifično takemu opravilu, na primer BPMN. S tako notacijo lahko namreč modeliramo tudi razne posebne elemente kot so časovne omejitve, kompenzacije, bolj specifične vloge, dogodki, tipi procesov ipd. Ko so poslovni procesi identificirani in modelirani na višjem nivoju abstrakcije z uporabo ArchiMate notacije, lahko nadaljujemo s konkretnjšim modeliranjem na nižjem nivoju tudi skozi več iteracij, v katerih se z vsako odkrije več detajlov o procesu samem.

Podobno je pri modeliranju tehničnih delov informacijsko-poslovne arhitekture. ArchiMate podpira identifikacijo aplikacij, njihovih vmesnikov in njihove uporabe v poslovnih procesih. Za modeliranje samega delovanja teh komponent in njihovih povezav pa je vsekakor bolj primeren kakšen drug jezik. Interna struktura aplikacij ter njihovo delovanje tudi sicer nista predmet poslovno-informacijske arhitekture temveč predmet njihove sistemske arhitekture. V ta namen obstaja veliko modelirnih jezikov izmed katerih je vsekakor najbolj razširjen UML. Tak nivo modeliranja je tudi sicer veliko bolj razvit in je standardna praksa razvoja informacijskih sistemov, UML pa že skoraj de facto standard za tovrstno modeliranje.

V domeni tehnološkega nivoja je zgodba enaka kot na aplikativnem oz. informacijskem nivoju. ArchiMate ima tudi zanj podporo na visokem nivoju, za natančnejše modeliranje povezovanja in delovanja tehnološke infrastrukture pa raje izberemo kaj drugega. UML je tudi v tem kontekstu izjemno pogost, saj vsebuje vse potrebne gradnike za tovrsten popis.

Usmerjenost ArchiMate-a v visoko nivojsko modeliranje pa ni njegova pomanjkljivost. Povsem nasprotno. Že zasnovan je bil tako, da so koncepti nižje nivojske arhitekture vseh treh slojev načrtno izpuščeni. Visokonivojski pogled je namreč točno to, za kar je bil ArchiMate ustvarjen. Samo s tega pogleda je namreč potreben pregled skozi vse sloje arhitekture. Ko se spuščamo nižje, postajajo sloji neodvisni. Izvedba določenega koraka v poslovnem procesu na primer ni odvisna od konkretne tehnične zasnove aplikacije, s pomočjo katere ga izvajamo. Podobno postavitve tehnične infrastrukture ni odvisna od programskih komponent aplikacije, temveč le od njene zunanje zasnove in uporabljene tehnologije. Ko torej preidemo na nivo, na katerem ni več soodvisnosti med različnimi sloji poslovno-informacijske arhitekture, ne potrebujemo več

enotnega jezika in tudi ne enotne ekipe, ki bi sodelovala pri tovrstnem načrtovanju. Na tej stopnji delo preide v ločene skupine in tudi ni več klasificirano kot razvoj poslovno-informacijske arhitekture.

## 5.2 Razširitve

Metode za poslovno-informacijsko arhitekturo, kot je na primer TOGAF, potrjujejo pomembnost modeliranja zahtev v procesu razvoja arhitekture. Možnost modeliranja je potrebna za specifikacijo, dokumentacijo, komunikacijo in tehtanje o ciljih in zahtevah. Tipične modelirne tehnike za poslovno-informacijsko arhitekturo poudarjajo produkte, storitve, procese in aplikacije organizacije. Dodatno so na razpolago še tehnike za opis strukturiranih seznamov zahtev in primerov uporabe. Malo pa je na voljo podpore za modeliranje motivacije, ki je podlaga za te zahteve in se pojavlja v obliki skrbi deležnikov in visokonivojskih ciljev, s katerimi naj bi te skrbi naslavljali [3].

Napisano je veljalo tudi za ArchiMate, dokler ni z verzijo 2.1 leta 2013 pridobil še motivacijsko razširitev ter implementacijsko in migracijsko razširitev. S tem je pridobil omenjeno manjkajočo podporo in širino, s katero lahko pokrije celotno vsebino TOGAF ogrodja.

Že v preliminarni fazi in nato v fazi A TOGAF zahteva določene izdelke, katerih vsebino bi bilo najprimerneje predstaviti grafično, vendar to z elementi iz jedra ArchiMate-a ni mogoče. Razširitve so bile zato ustvarjene specifično za podporo tem opravilom.

Kot rečeno je del preliminarne faze organizacijski model, tudi znotraj izjave o poslovnih principih, ciljih in gonilu pa bi marsikatera predstavitev bila jasnejša v obliki modela. Faza A nato predvideva identifikacijo deležnikov, zahtev in pomislekov, ki bi jih lahko predstavili podobno kot elemente iz preliminarne faze.

V ta namen je nastala motivacijska razširitev.

Motivacijska razširitev vsebuje koncepte, ki jih srečujemo v kontekstu razlogov za vpeljavo poslovno-informacijske arhitekture:

- Deležnik (stakeholder)
- Gonilo (driver)
- Ocena (assessment)
- Cilj (goal)
- Zahteva (requirement)

- Omejitev (constraint)
- Princip (principle)

TOGAF faze E, F in G prav tako zahtevajo določene izdelke, za katere elementi jedra ArchiMate-a niso primerni ali pa niso zadostni. Vsebujejo celovito analizo vrzeli ter planiranje delovnih nalog in njihovo izvedbo, za kar je bila ustvarjena implementacijska in migracijska razširitev.

Implementacijska in migracijska razširitev tako vsebuje koncepte, ki jih srečujemo v kontekstu implementacije nove arhitekture in migracije iz obstoječe na ciljno arhitekturo:

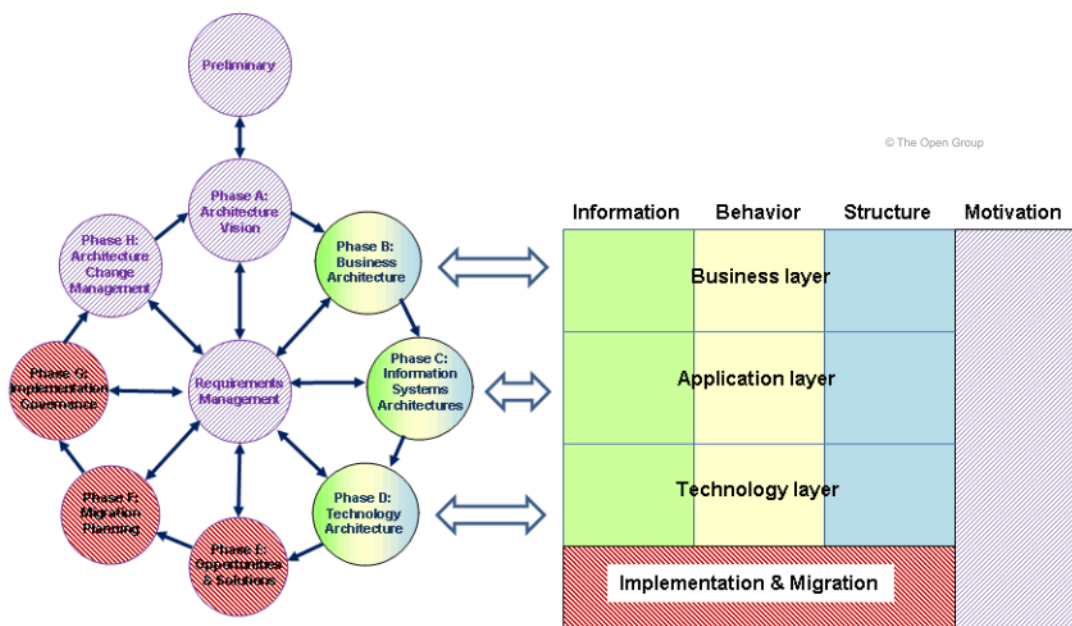
- Delovni sklop (Work package)
- Izdelek (Deliverable)
- Plato (Plateau)
- Vrzeli (Gap)

## 5.3 Souporaba s TOGAF-om

Tako ArchiMate kot TOGAF se razvijata pod okriljem The Open Group, zaradi česar njun razvoj poteka zelo skladno in drug drugega podpirata. Tudi njuni začetki izvirajo iz istih temeljnih konceptov in imata zato zelo podoben meta model. Sta si pa različna z vidika primarnega namena. TOGAF poudarja metodologijo, ArchiMate prezentacijo. Ta komplementarnost pa je ravno to, kar ju naredi idealna partnerja pri izvajanju poslovno-informacijske arhitekture. Oba sta tudi izjemno razširjena in zanju obstaja vrsta literature in drugih virov, ki organizacijam lajšajo njuno uporabo.

Že na prvi pogled lahko potegnemo vzporednice med TOGAF-ovimi fazami B, C in D (poslovna arhitektura, informacijska arhitektura, tehnološka arhitektura) in ArchiMate-ovimi ravnmi (poslovna raven, aplikacijska raven, tehnološka raven). Dolgo se je zato tako kombinacijo uporabljalo samo v teh fazah. Razširitve v zadnji verziji ArchiMate-a pa so tudi v to ogrodje prinesle vsebino, s katero lahko pokrijemo preostale faze ADM metodologije.





Slika 5-2: Preslikava ogrodij ArchiMate in TOGAF drugega na drugo[10]

Ob bolj podrobnem pogledu ugotovimo, da tudi ob taki povezavi nekateri TOGAF-ove entitete niso pokrite. Vendar zanje lahko navadno uporabimo kombinacijo elementov ali dodatne karakteristike.

Podobno velja za vidike. Obe ogrodji imata definiran seznam tipičnih vidikov, ki bi jih bilo treba pokriti. Nemogoče jih je sicer povezati po principu 1:1, vendar je ArchiMate vseeno sposoben vsebinsko pokriti vse TOGAF-ove zahteve.

Če v TOGAF meta modelu začnemo z razdelkom “Arhitekturna vizija, principi in zahteve”, v njem identificiramo spodnje entitete, ki jih lahko tudi povežemo z ArchiMate-ovimi koncepti.

#### ***TOGAF entiteta    ArchiMate koncept    Izvor***

|              |                       |   |
|--------------|-----------------------|---|
| princip      | princip               | motivacijska razširitev                                   |
| omejitev     | omejitev              | motivacijska razširitev                                   |
| predpostavka | ni direktne pretvorbe | lahko jih upodobimo s pomočjo atributa na drugem konceptu |
| zahteva      | zahteva               | motivacijska razširitev                                   |
| vrzel        | vrzel                 | implementacijska in migracijska razširitev                |

|               |                       |   |
|---------------|-----------------------|---|
| delovni paket | delovni paket         | implementacijska in migracijska razširitev                |
| zmožnost      | ni direktne pretvorbe | lahko jih upodobimo s pomočjo grupiranja drugih konceptov |

*Tabela 5-1: Preslikava entitet iz skupine "Arhitekturna vizija, principi in zahteve"*

V razdelku poslovne arhitekture srečamo največje število entitet. Le-te in njihove pripadajoče ArchiMate ekvivalente najdemo v tabeli 5-2: Preslikava entitet iz skupine "Poslovna arhitektura".

| <b><i>TOGAF entiteta</i></b> | <b><i>ArchiMate koncept</i></b>         | <b><i>Izvor</i></b>   |
|------------------------------|---|---|
| vodilo                       | vodilo                                  | motivacijska razširitev   |
| cilj                         | cilj                                    | motivacijska razširitev   |
| namen                        | cilj                                    | motivacijska razširitev   |
| merilo                       | ni direktne pretvorbe                   | izrazimo ga lahko kot atribut cilja ali kakšnega drugega koncepta |
| organizacijska enota         | poslovni akter                          | jedro, poslovna raven   |
| akter                        | poslovni akter, aplikacijska komponenta | jedro, poslovna/aplikacijska raven                                |
| vloga                        | poslovna vloga                          | jedro, poslovna raven   |
| lokacija                     | lokacija                                | jedro, poslovna raven   |
| funkcija                     | poslovna funkcija                       | jedro, poslovna raven   |
| poslovna storitev            | poslovna storitev                       | jedro, poslovna raven   |
| pogodba                      | pogodba                                 | jedro, poslovna raven   |
| kakovost storitve            | ni direktne pretvorbe                   | izrazimo jo lahko kot atribut storitve                            |
| proces                       | poslovni proces                         | jedro, poslovna raven   |
| dogodek                      | poslovni dogodek                        | jedro, poslovna raven   |

|            |                       |   |
|------------|-----------------------|---|
| krmiljenje | ni direktne pretvorbe | lahko ga upodobimo kot poslovno funkcijo ali proces |
| produkt    | produkt               | jedro, poslovna raven                               |

*Tabela 5-2: Preslikava entitet iz skupine "Poslovna arhitektura"*

Kot vidimo je večina pripadajočih ArchiMate konceptov iz poslovne ravni, kar jasno nakazuje na povezavo med to ravno in fazo B ADM metode.

Razdelek informacijske arhitekture vsebuje entitete podatkovne in aplikacijske arhitekture. Le-te in njihove pripadajoče ArchiMate ekvivalente najdemo v tabeli 5-3: Preslikava entitet iz skupine "Informacijska arhitektura".

| <b><i>TOGAF entiteta</i></b>     | <b><i>ArchiMate koncept</i></b> | <b><i>Izvor</i></b>        |
|----------------------------------|---------------------------------|----------------------------|
| podatkovna entiteta              | podatkovni objekt               | jedro, informacijska raven |
| logična podatkovna komponenta    | podatkovni objekt               | jedro, informacijska raven |
| fizična podatkovna komponenta    | artefakt                        | jedro, informacijska raven |
| storitev informacijskega sistema | aplikacijska storitev           | jedro, informacijska raven |
| logična aplikacijska komponenta  | aplikacijska komponenta         | jedro, informacijska raven |
| fizična aplikacijska komponenta  | aplikacijska storitev           | jedro, informacijska raven |

*Tabela 5-3: Preslikava entitet iz skupine "Informacijska arhitektura"*

Vsi pripadajoči ArchiMate koncepti so iz informacijske ravni jedra ArchiMate-a. To kaže na jasno povezavo med to ravno in fazo C ADM metode.

Ostaja nam le še razdelek tehnološke arhitekture. Njegove entitete in njihove pripadajoče ArchiMate ekvivalente najdemo v tabeli 5-4: Preslikava entitet iz skupine "Tehnološka arhitektura".

| <b><i>TOGAF entiteta</i></b>  | <b><i>ArchiMate koncept</i></b> | <b><i>izvor</i></b>     |
|-------------------------------|---------------------------------|-------------------------|
| storitev platforme            | infrastruktura storitev         | jedro, tehnološka raven |
| logična tehnološka komponenta | vozlišče, komunikacijska pot    | jedro, tehnološka raven |

|                               |   |                         |
|-------------------------------|---|-------------------------|
| fizična tehnološka komponenta | naprava, omrežje, sistemska programska oprema | jedro, tehnološka raven |
|-------------------------------|---|-------------------------|

*Tabela 5-4: Preslikava entitet iz skupine "Tehnološka arhitektura"*

Vsi pripadajoči ArchiMate koncepti so iz tehnološke ravni jedra ArchiMate-a. To ponovno kaže na jasno povezavo med to ravnjo in fazo D ADM metode.

Torej, TOGAF-ove faze B, C in D (poslovna arhitektura, informacijska arhitektura, tehnološka arhitektura) lahko preprosto prevedemo v poslovno, aplikacijsko in tehnološko raven ArchiMate ogrodja. Faze E, F in H (priložnosti in rešitve, planiranje migracije in upravljanje implementacije) pokriva implementacijska in migracijska razširitev, medtem ko lahko z motivacijsko razširitvijo pokrijemo fazi A in H (arhitekturna vizija in upravljanje z arhitekturnimi spremembami) ter preliminarno fazo in upravljanje z zahtevami [10].

## 6 Frameworx

Frameworx je paket standardov, ki ga razvija TM Forum, neprofitno združenje ponudnikov storitev telekomunikacijske industrije. Kakor TOGAF in ArchiMate, je tudi Frameworx ogrodje za poslovno-informacijsko arhitekturo, vendar je namenjeno specifično telekomunikacijski industriji ter sorodnim storitvenim podjetjem. Tvorijo ga trije sklopi, eTOM, SID in TAM, vse tri pa povezuje Integration framework (integracijsko ogrodje).

eTOM ali Business Process Framework je model oz. ogrodje poslovnih procesov za uporabo v informacijski, komunikacijski in zabavni industriji. Poudarek je na poslovnih procesih, ki jih uporabljajo ponudniki storitev, na povezavah med temi procesi, identifikaciji vmesnikov in uporabi informacij o strankah, storitvah, resursih ter partnerjih oz. dobaviteljih. Je referenčno ogrodje za kategorizacijo vseh poslovnih aktivnosti, ki jih ponudnik storitev izvaja in sicer na strukturiran način, ki dovoljuje pregled na različnih nivojih podrobnosti. Ponudnikom storitev lahko eTOM služi kot načrt za usmerjanje procesov in nudi nevtralno referenco za prenovu tako internih procesov kot procesov, ki podpirajo delo z zunanjimi partnerji.

SID ali Information framework (informacijsko ogrodje) vsebuje standardne definicije za vse informacije, ki tečejo znotraj podjetja ter med ponudniki storitev in njihovimi poslovnimi partnerji. Z uporabo SID-a se izognemo nesporazumom, saj prinaša skupni slovar in referenčni model v obliki entitet in razmerij med njimi, ki nosijo informacije, ki jih potrebujemo za implementacijo eTOM-a.

TAM ali Application framework (aplikacijsko ogrodje) je načrt vseh sistemov, ki zajema, kako so poslovni procesi implementirani v aplikacijah. Poleg tega opisuje funkcionalnost posameznih aplikacij in tako vsebuje tudi skupen jezik, ki naj bi ga kupci in dobavitelji uporabljali za učinkovito sodelovanje.

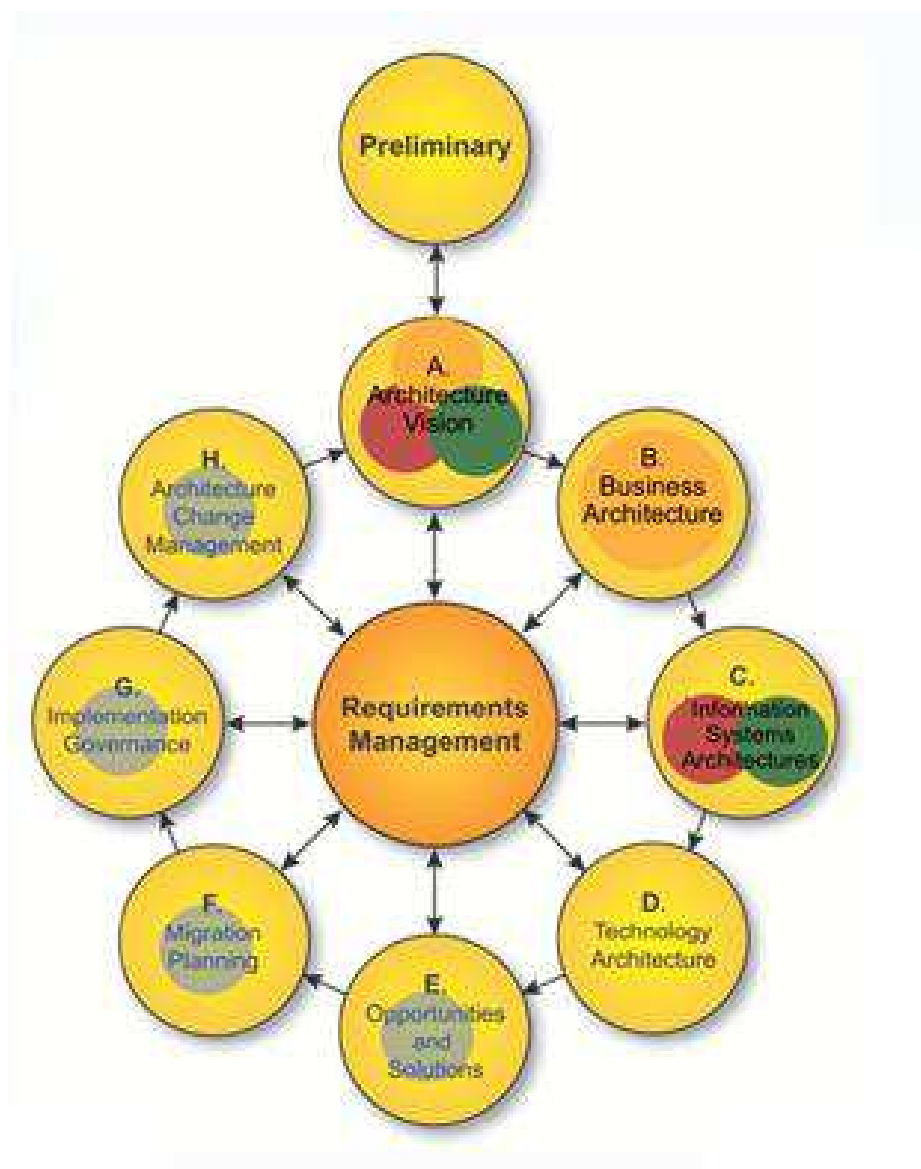
Integration framework je vrsta standardov, ki podpira interoperabilnost med aplikacijami, definiranimi v TAM-u z vmesniki, definiranimi s pomočjo entitet SID-a in zahtevami, ki se zberejo na podlagi procesov iz eTOM-a. Z njim se kot že rečeno vse enote Frameworx-a povežejo v celoto.

Da bi lahko razumeli arhitekturo določenega ponudnika storitve v določenem časovnem obdobju, je potrebno upoštevati vse objekte, ki se spreminjajo in način, s katerim izvajamo transformacijo ter akterje, ki so zanjo odgovorni. Sam Framework ne vsebuje teh informacij, temveč jih je potrebno definirati kot poslovno strategijo v obliki poslovnih zahtev [6]. Vsebuje pa orodja za pripravo teh informacij.

V telekomunikacijski industriji je uporaba specifičnih referenčnih modelov bistvena za harmonizacijo poslovnih procesov in informacijske tehnologije [1]. In od preostalih ogrodij se Framework razlikuje po tem, da predstavlja predvsem referenčni model. Framework sam ne vsebuje ne metodologije ne notacije za predstavitev arhitekture, zato bi ga pri praksi poslovno-informacijske arhitekture težko uporabljali samega. V svoji referenčni vlogi pa je lahko komplementaren drugim ogrodjem, na primer TOGAF-u in ArchiMate-u. V taki kombinaciji TOGAF predstavlja centralno entiteto, ki povezuje ostali dve ogrodji. Kot že rečeno v taki kombinaciji TOGAF predstavlja metodologijo, ArchiMate jezik za reprezentacijo, Framework pa lahko na drugi strani TOGAF-u dodamo v smislu vsebine, nad katero se metodologija izvaja. Trije Frameworkovi sklopi lahko v grobem pripomorejo k delu v fazah A, B in C (slika 6-1: Uporaba eTOM, SID in TAM v fazah ADM [6]).

Framework lahko delno pokriva fazo A z naborom definicij povezanih s strategijo organizacije. Zato je lahko v pomoč pri začetni opredelitvi ciljne arhitekture. eTOM opisuje poslovne procese ponudnikov storitev in ga lahko uporabimo v fazi B ADM metodologije za opredelitev poslovne arhitekture. SID je model informacij in podatkov in ga kot takega lahko uporabimo v fazi C ADM metodologije kot pomoč pri opredelitvi podatkovne arhitekture. TAM je ogrodje aplikacij, ki jih uporablja tipičen ponudnik storitev. Uporablja se ga za klasifikacijo celotnega nabora aplikacij in ga kot takega lahko uporabimo v fazi C ADM metodologije kot pomoč pri opredelitvi aplikacijske arhitekture.

S tem zapolnimo še zadnjo vrzel in kombiniramo tri izjemno razširjena ogrodja v celovit nabor pripomočkov za izvajanje poslovno-informacijske arhitekture.



Slika 6-1: Uporaba eTOM, SID in TAM v fazah ADM [6]





## 7 Arhitekturne dobre prakse

Ugotovili smo, da sta TOGAF in ArchiMate idealen par pri izvajanju poslovno-informacijske arhitekture. Vendar kljub širini, ki jo pokrivata, ne obravnavata vsebine. Tako je tudi prav. Pri souporabi TOGAF doprinese metodologijo, ArchiMate notacijo, koncepti so v glavnem skupni, ne eden ne drugi pa ne definira, kakšna naj opisana arhitektura bo. Vsebina je torej popolnoma prepuščena izvajalcem.

Ob uvajanju poslovno-informacijske arhitekture je navadno organizacija že razmeroma zrela in ima posledično že zelo razvite vse tri ravni arhitekture. Pri praksi poslovno-informacijske arhitekture je torej nikoli ne postavljamo znova, temveč vedno začnemo z opisom obstoječe. V tem primeru je vsebina znana, potrebno je samo zbrati dovolj ljudi z dovolj znanja o obstoječem stanju ter izvesti popis.

Ciljna arhitektura pa je neobstoječa. Popolnoma je prepuščena arhitektom, ki jo morajo zasnovati na podlagi ciljev in zahtev ter seveda na podlagi obstoječe arhitekture in zmožnosti prehoda na ciljno. Poleg lastnih izkušenj so za ta korak arhitektom lahko v pomoč tudi arhitekturne dobre prakse, torej vzorci in smernice, ki so že bile večkrat uporabljene v drugih organizacijah in so se izkazale za uspešne.



## 8 Preliminarna faza in arhitekturna vizija

Preliminarna faza nastopi z odločitvijo o uvedbi prakse poslovno-informacijske arhitekture, medtem ko je arhitekturna vizija že prvi korak v samem življenjskem ciklu arhitekture in se z vsako iteracijo ponovi.

V sklopu preliminarne faze izvajamo naslednje korake:

- Ocena obsega vpliva na organizacijo
- Izbira ogrodij za upravljanje in podporo
- Vzpostavitev ekipe za poslovno-informacijsko arhitekturo in njena umestitev v organizacijsko strukturo
- Identifikacija arhitekturnih principov
- Prilagoditev TOGAF-a in drugih izbranih ogrodij
- Implementacija arhitekturnih orodij

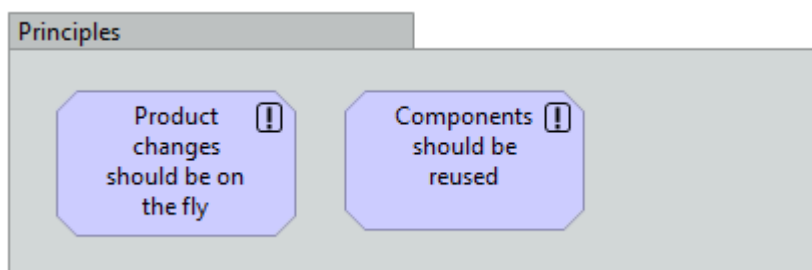
Večina aktivnosti torej predstavlja vzpostavitev prakse in pripravo na vzpostavitev projekta prve iteracije. Glavni izdelek pa predstavlja katalog principov.

Ob koncu preliminarne faze naj bi tako imeli:

- Organizacijski model za poslovno-informacijsko arhitekturo (ocena obsega vpliva na organizacijo, ocena zrelosti, vloge in odgovornosti, omejitve pri arhitekturnem delu, proračunske zahteve):
  - omejitve in proračunske zahteve lahko opredelimo v okviru modela motivacijske ravni: slika 8-2: Model motivacijske ravni
  - obseg vpliva je izjemno širok, saj pokriva procese organizacijskih enot marketinga, prodaje, naročniškega sektorja in IT-ja. Natančnejša ocena bo na voljo v fazi B ob popisu dejanskih poslovnih procesov, na katere ima vpliv sprememba v arhitekturi.
  - Ocena zrelosti je lahko zelo subjektivna. Ker gre za prvo iteracijo življenjskega cikla poslovno-informacijske arhitekture, lahko privzamemo, da je pripravljenost relativno nizka ter da je na voljo razmeroma malo dokumentacije

na temo obstoječe postavitve. Vloge in odgovornosti še niso postavljene, to pa je potrebno storiti v okviru koraka »vzpostavitev ekipe za poslovno-informacijsko arhitekturo«.

- Prilagoditev arhitekturnega ogrodja (prilagojena arhitekturna metoda, prilagojena arhitekturna vsebina, arhitekturni principi, uporabljena orodja): izbrana arhitekturna metoda je ADM. Ker pa ne gre za realen projekt, temveč le za potrebe seminarske naloge, se določeni koraki kot so odobritve, vzpostavitev ekipe ipd. izpuščajo.
- Začetni arhitekturni repozitorij: v začetni repozitorij se na tem koraku doda le izbrano ogrodje, torej TOGAF.
- Zahteva za arhitekturno delo: uraden dokument, ki vzpostavlja projekt.
- Ogrodje za upravljanje arhitekture: spet TOGAF.
- Katalog principov:



*Slika 8-1: Katalog principov*

Preliminarni fazi in nato tudi fazi A je potrebno nameniti posebno pozornost, saj zbiranje zahtev in način, kako njihova realizacija vpliva na zadane cilje, bistveno vpliva na vse nadaljnje faze. Predvsem vpliv na cilje je v projektih prenove velikokrat zanemarjen korak. Ob koncu implementacije pa se potem poraja vprašanje, zakaj kljub uspešnosti projekta samega rezultati niso skladni s pričakovanji.

TOGAF priznava, da so cilji in zahteve osrednje gonilo pri procesu razvoja arhitekture. Po ADM metodi je upravljanje z zahtevami osrednji proces, ki se ga izvaja skozi vse faze cikla. Sposobnost spopadanja s spreminjajočimi zahtevami je za samo metodologijo ključna, saj je arhitektura po naravi podvržena negotovosti in spremembam ter poskuša premostiti razkorak med težnjami deležnikov in praktično dosegljivimi rešitvami.

Modeliranje zahtev pomaga pri razumevanju, strukturiranju in analizi relacije med poslovnimi in tehničnimi zahtevami ter s tem lajša usklajevanje med poslovanjem in IT-jem. Koncept cilja v ciljno orientiranih zahtevah se na primer uporablja za definicijo zelenega učinka, ki si ga želi določen deležnik. Ta cilj je lahko povezan z enim ali več poslovnimi cilji, ki definirajo, zakaj

je cilj potreben ter je povezan z bolj konkretnimi tehničnimi cilji, ki definirajo, kako naj bo cilj realiziran [3].

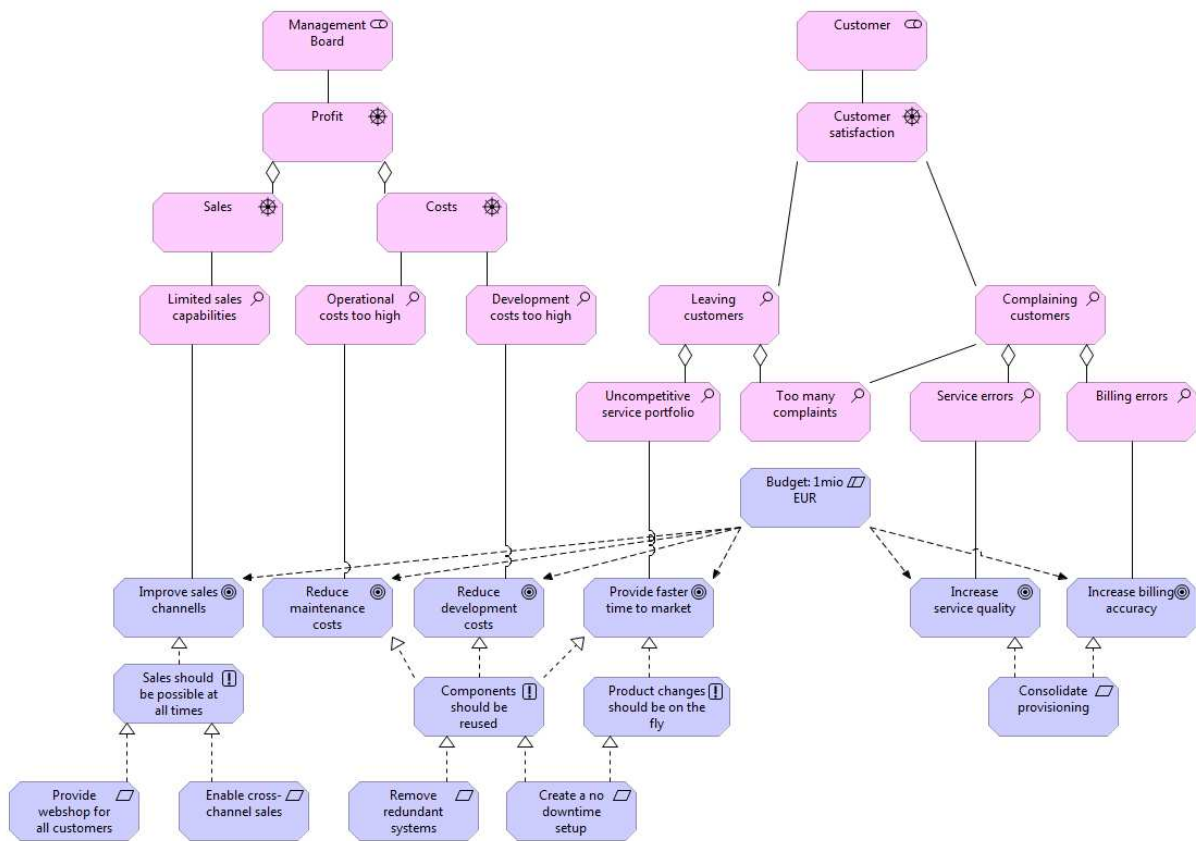
Faza A predstavlja arhitekturno vizijo. V njenem okviru izvajamo naslednje korake:

- Uradna vzpostavitev arhitekturnega projekta
- Identifikacija deležnikov, skrbi in poslovnih zahtev
- Potrditev in dodelava poslovnih ciljev, vodil in skrbi
- Evalvacija poslovnih zmožnosti
- Ocena pripravljenosti na poslovno transformacijo
- Definicija obsega
- Potrditev in dodelava arhitekturnih in poslovnih principov
- Razvoj arhitekturne vizije
- Definicija vrednosti ciljne arhitekture ter njeni ključni kazalniki uspeha
- Identifikacija tveganj poslovne transformacije ter aktivnosti za njihovo zmanjševanje
- Razvoj izjave o arhitekturnem delu ter njegova odobritev

Rezultati faze A so nato sledeči:

- Odobrena izjava o arhitekturnem delu (projektna definicija in obseg, pregled arhitekturne vizije, projektni plan in časovnica)
  - Projektna definicija in obseg sta bila podana v uvodnem delu seminarske naloge. Na kratko gre za vzpostavitev nove aplikacijske arhitekture na predelu BSS sistemov ter prilagoditev poslovne in tehnološke arhitekture, ki jo to prinese.
  - Arhitekturno vizijo lahko povzamemo z modelom motivacijske ravni, ki sledi v naslednji točki
  - Ker ne gre za realni projekt, na tem mestu projektni plan in časovnica nista bila izdelana
- Dodelana izjava o poslovnih principih, ciljih in vodilih: slika 8-2: Model motivacijske ravni
- Arhitekturni principi: slika 8-2: Model motivacijske ravni

Tako preliminarno fazo kot arhitekturno vizijo pokriva motivacijska raven ArchiMate-a. Njeni ključni koncepti so deležnik, vodilo, ocena, cilj, zahteva, omejitev in princip. Z njimi lahko v enem modelu ponazorimo tako dodelano izjavo o poslovnih principih, ciljih in vodilih kot arhitekturne principe.



Slika 8-2: Model motivacijske ravni

## 9 Poslovna arhitektura

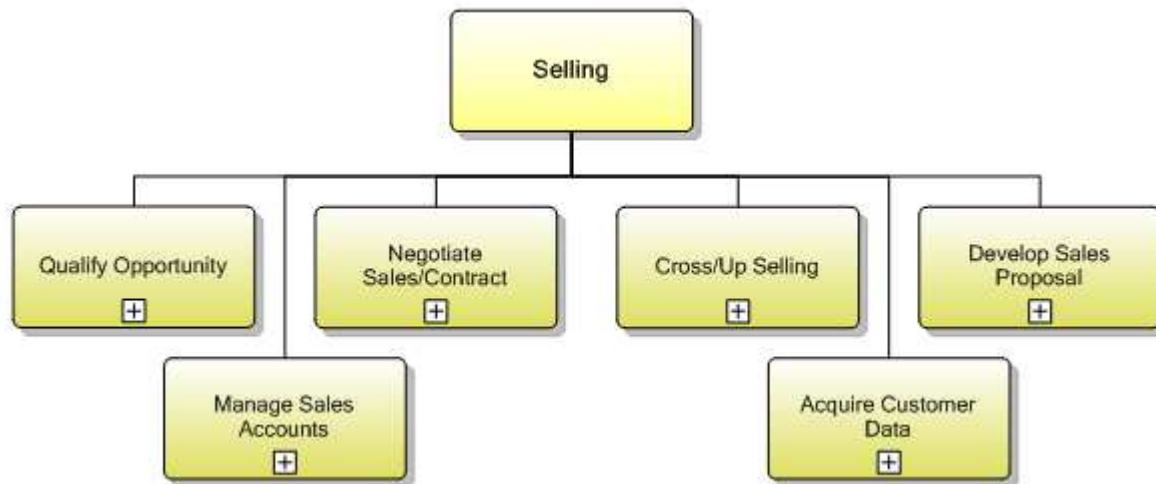
Faza B – Poslovna arhitektura je prva faza dejanske opredelitve arhitekture same. Tekom nje je potrebno pripraviti trenutno in ciljno poslovno arhitekturo ter analizo vrzeli med njima. Za modeliranje vseh treh uporabimo jedro ArchiMate ogrodja in sicer vse tri strukture na poslovni ravni.

Popis vseh procesov, dogodkov, akterjev ter ostalih konceptov poslovne arhitekture pa je lahko izjemno razširjen. Ker se nato informacijska in tehnična arhitektura navezujeta nanjo, se ta kompleksnost z vsako fazo le še stopnjuje. Za potrebe te naloge bomo zato predstavili le določene ključne procese, v katerih so udeleženi sistemi za podporo poslovanju.

Za izbiro teh procesov si pomagamo z eTOM modelom. Že na prvi pogled je jasno, da domene angažirana stranka (Engaged party), vir (Resource) in storitev (Service) ne bodo relevantne. Ti procesi namreč uporabljajo bodisi sisteme za splošne službe ali OSS sisteme.

Relevantne domene so tako stranka (Customer), produkt (Product) ter trženje/prodaja (Marketing/Sales). V strankini domeni tako ostajajo samo operativni procesi, iz produktne domene je velika večina procesov na strani strategije, infrastrukture in produkta in ti bodo za nas tudi najbolj zanimivi. Iz domene trženja in prodaje bodo najbolj zanimivi procesi na meji med eno in drugo stranjo.

Začnimo torej s trženjem in prodajo.

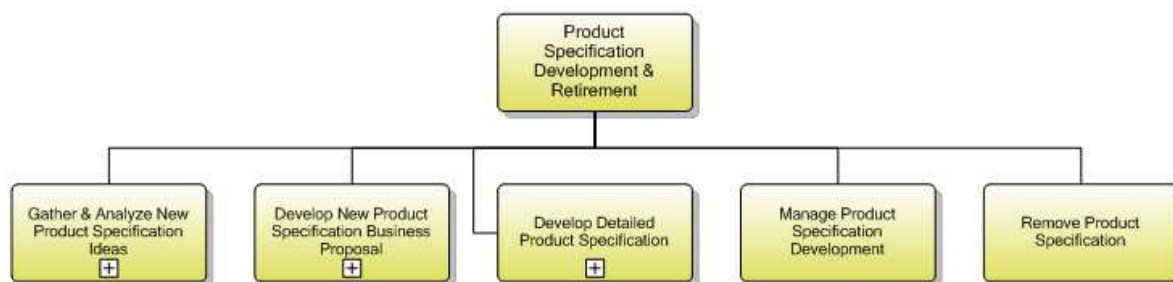


Slika 9-1: eTOM razdelek 1.1.9 Prodaja [15]

eTOM razdelek 1.1.9 Prodaja, je vsekakor ključni proces v delovanju ponudnika storitev in pri njem sistemi za podporo poslovanju igrajo veliko vlogo. Je pa pomemben tudi zato, ker predstavlja korak, ki se izvaja pred ravnanjem z naročili, ki ga bomo prav tako obravnavali. V sklopu naših izbranih ciljev so najpomembnejši podprocesi iz tega sklopa 1.1.9.3 Prodaja komplementarnih ali dodatnih storitev (Cross/Up Selling), 1.1.9.6 Pridobivanje podatkov o stranki (Acquire Customer Data) in 1.1.9.4 Priprava prodajne ponudbe (Develop Sales Proposal).

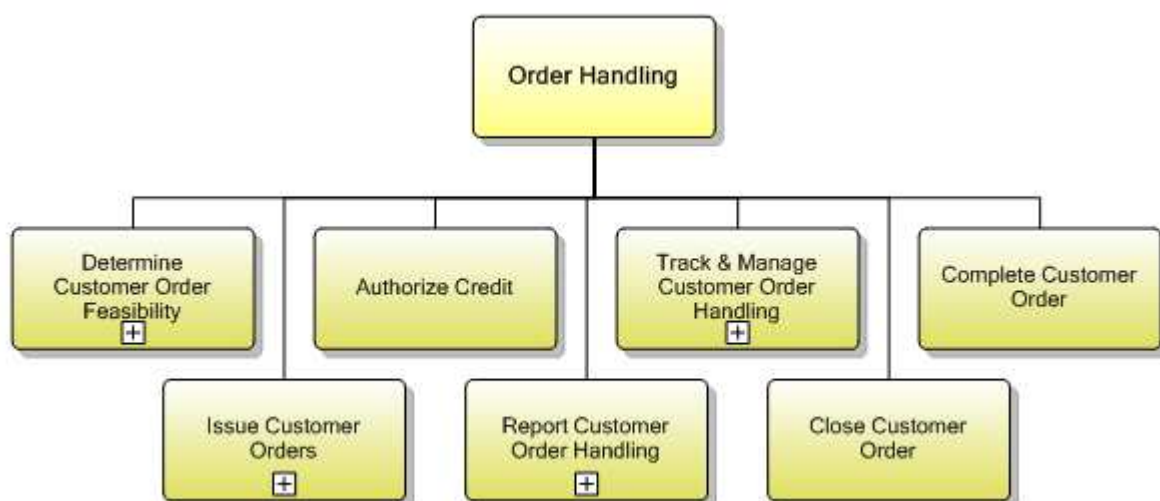
V produktni domeni je za nas najpomembnejši proces izdelave novih produktov. Specifično bi radi spremenili podprocesa 1.2.7.1 Razvoj in umik produktnih specifikacij ter 1.2.7.2 Razvoj in umik produktnih ponudb. Ta dva sta namreč zelo odvisna od sistemov, na katerih je potrebno produkte in ponudbe razviti in to so sistemi za podporo poslovanju. Obravnavali ju bomo kar skupaj, saj sta del 1.2.7 Razvoj in umik produktnih specifikacij in ponudb, njuni rezultati koristijo enake sisteme ter imajo enak vpliv na kasnejši korak 1.2.5 Upravljanje s konfiguracijo produktov.





Slika 9-2: eTOM razdelek 1.2.7.1 Razvoj in umik produktnih specifikacij [15]

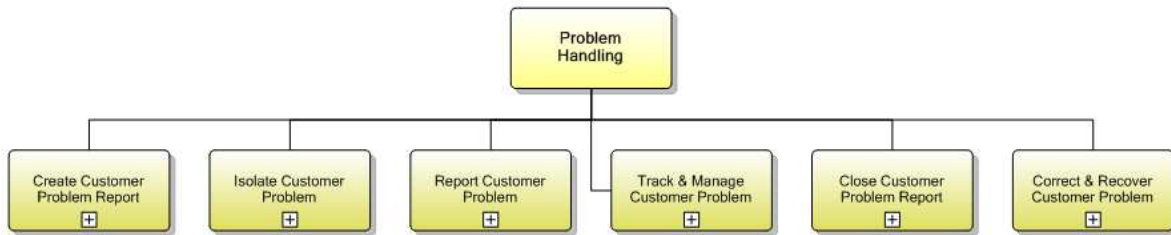
Praktično vsi procesi iz strankine domene so relevantni za kontekst te naloge, vendar se bomo omejili na tri.



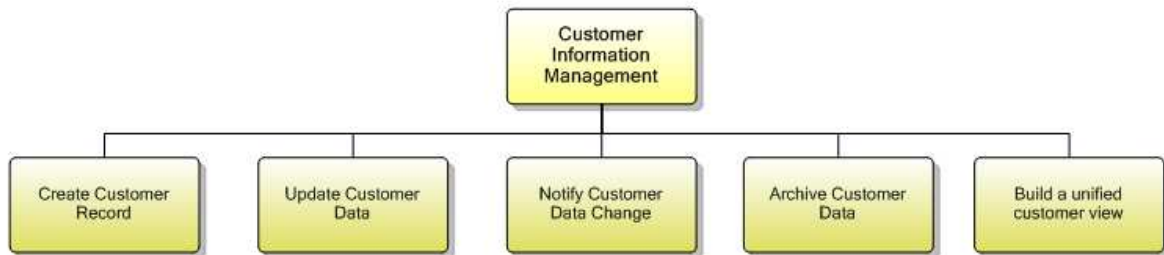
Slika 9-3: eTOM razdelek 1.3.3 Ravnanje z naročili [15]

Ravnanje z naročili je definitivno najpomembnejši med njimi, saj bodisi posredno ali neposredno vpliva na praktično vse cilje, ki smo jih definirali. Vpliva tako na izboljšanje prodajnih kanalov kot na stroške vzdrževanja in razvoja, na izboljšanje potrebnega časa za vpeljavo novega produkta ter na izboljšanje kvalitete storitev in natančnosti obračunavanja storitev.

Poleg ravnanja z naročili bomo obravnavali še ravnanje s težavami ter upravljanje s podatki o stranki.



Slika 9-4: eTOM razdelek 1.3.7 Ravnanje s težavami [15]



Slika 9-5: eTOM razdelek 1.3.6 Upravljanje s podatki o stranki [15]

Za izbran obseg moramo torej izvesti predvidene korake, ki jih vsebuje faza B:

- Izбира referenčnih modelov, zornih kotov in orodij: kot referenčni model smo izbrali eTOM, posebnih zornih kotov ne bomo uporabljali, ker smo omejili število poslovnih procesov, ki jih obravnavamo in lahko zato prikažemo celoten pogled nanje, kot orodje za modeliranje pa smo že v prejšnji fazi in bomo tudi v naslednjih uporabljali Archi [20]
- Priprava opisa trenutne poslovne arhitekture
- Priprava opisa ciljne poslovne arhitekture
- Izdelava analize vrzeli med njima
- Definicija kandidatov za komponente načrta: tu se pripravi grob načrt za prehod, ki ga nato natančneje definiramo v fazi E – Implementacija arhitekture. Sprejete odločitve so tako prikazane v poglavju 12 Implementacija arhitekture.
- Razrešitev posledic na celotno arhitekturo: nova poslovna arhitektura ne bo imela večjih posledic na organizacijo (primer večje posledica bi bila potreba po spremembi organizacijske strukture, potreba po dodajanju novih kadrov ali odpuščanju starih)
- Formalni pregled z deležniki
- Finalizacija poslovne arhitekture
- Dodajanje izdelkov v dokument arhitekturne definicije

V sklopu faze B za izbrani obseg TOGAF predvideva naslednje izdelke:

- Katalog organizacija/akter

- Katalog vodilo/cilj/namen
- Katalog vlog
- Katalog poslovna storitev/funkcija
- Katalog lokacij
- Katalog proces/dogodek/nadzor/produkt
- Katalog pogodba/merilo
- Matrika poslovnih sodelovanj
- Matrika akter/vloga
- Diagram poslovnih sledi
- Diagram poslovna storitev/informacija
- Diagram funkcionalne dekompozicije
- Diagram produktnega življenjskega cikla
- Diagram cilj/namen/storitev
- Diagram poslovnih primerov uporabe
- Diagram dekompozicije organizacije
- Diagram procesnih tokov
- Diagram dogodkov

Seznam je kar obsežen, vendar je to le seznam možnih ali priporočenih izdelkov, ki naj bi nastali tekom te faze. V praksi se glede na vsebino, ki jo pokrivamo, omejimo na tiste izdelke, s katerimi lahko dosežemo maksimalno razmerje med vloženim delom in pokritostjo ter jasnostjo prezentacije.

V ta namen začnemo z diagramom procesnih tokov, ki vsebuje večino gradnikov, ki so uporabljeni na zgornjem seznamu. Tako lahko hkrati pokrijemo veliko katalogov in diagramov, saj za njihovo predstavitev spremenimo le pogled.

## 9.1 Trenutna poslovna arhitektura

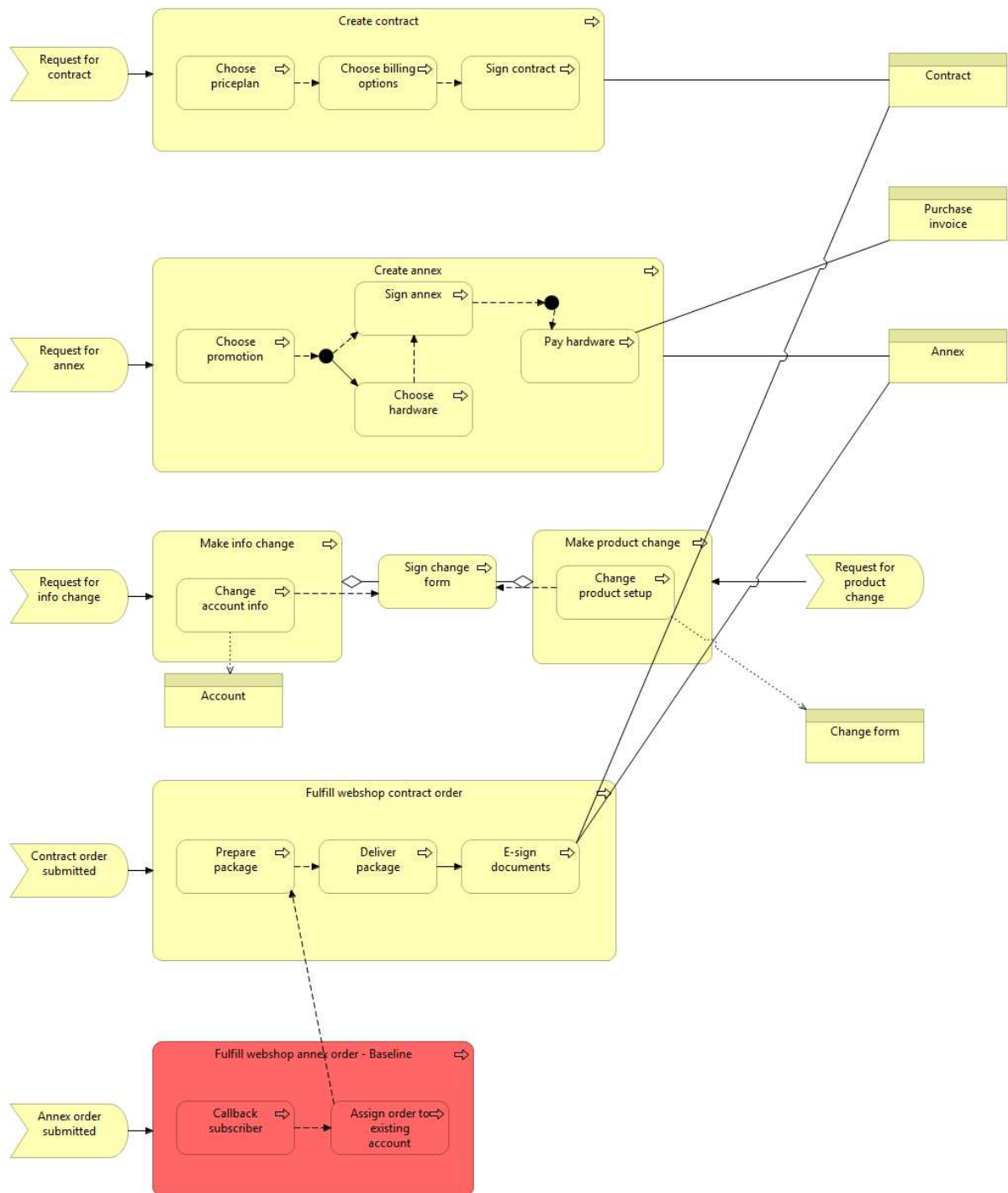
V trenutni poslovni arhitekturi so glavni procesi, ki so predmet sistemov za podporo poslovanju in pokrivajo izbrane eTOM elemente, naslednji:

- Sklepanje pogodbe
- Sklepanje aneksa na pogodbo
- Spreminjanje podatkov o naročniku
- Spreminjanje naročnikovih produktov

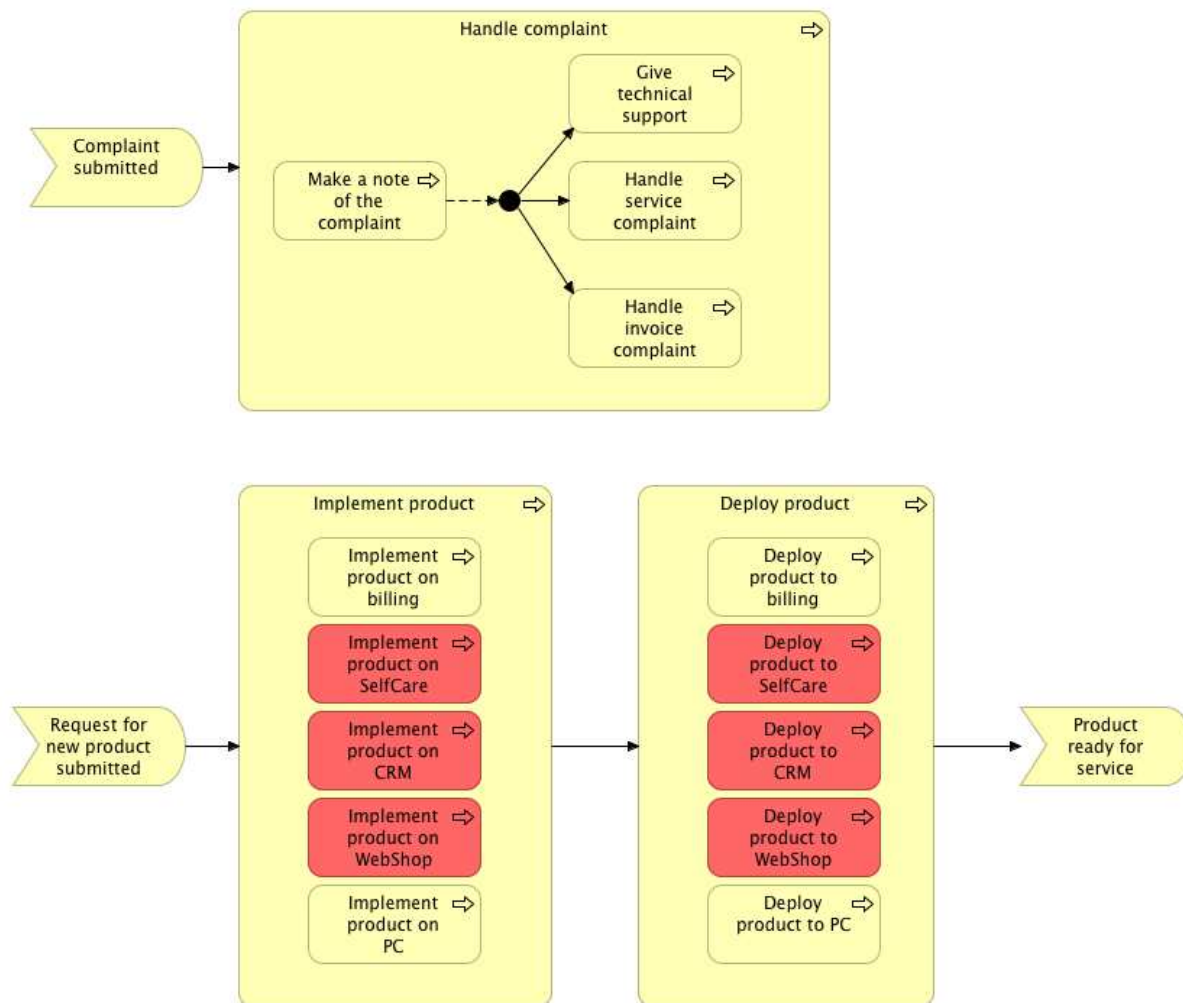
- Obravnavanje pritožb
- Uvedba novega produkta

Sklepanje pogodbe je proces, ki ga podpirata že omenjena procesa prodaje in upravljanja z naročili, dodatno pa še upravljanje s stranko. Sklepanje aneksa na pogodbo ravno tako pokrivata prodaja in upravljanje z naročili, enako velja za spreminjanje naročnikovih produktov. Spreminjanje podatkov o naročniku je predmet upravljanja podatkov o stranko. Obravnavanje pritožb pokriva sklop ravnanja s težavami. Uvedbo novega produkta pa pokriva razvoj in umik produktnih specifikacij in ponudb.

Njihovi podprocesi, dogodki, ki jih prožijo ter objekti, nad katerimi se izvajajo, so prikazani na spodnjih slikah.



Slika 9-6: Trenutna poslovna arhitektura – 1

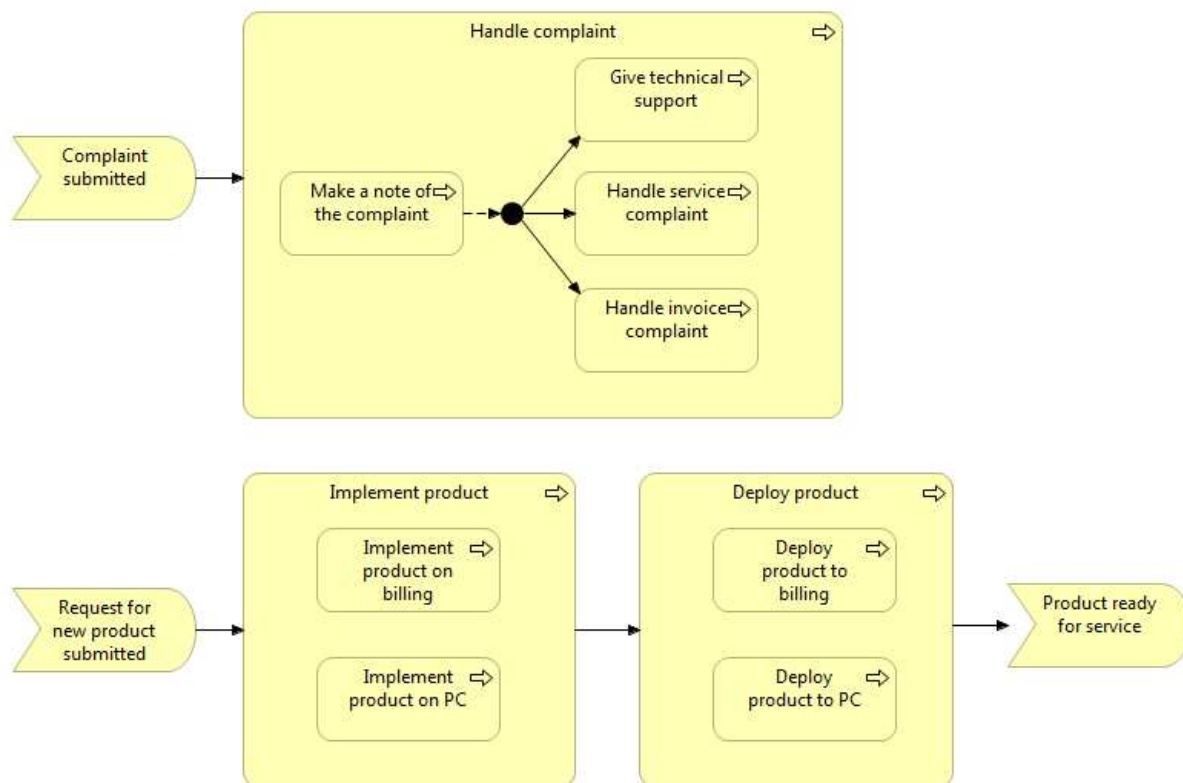


Slika 9-7: Trenutna poslovna arhitektura – 2

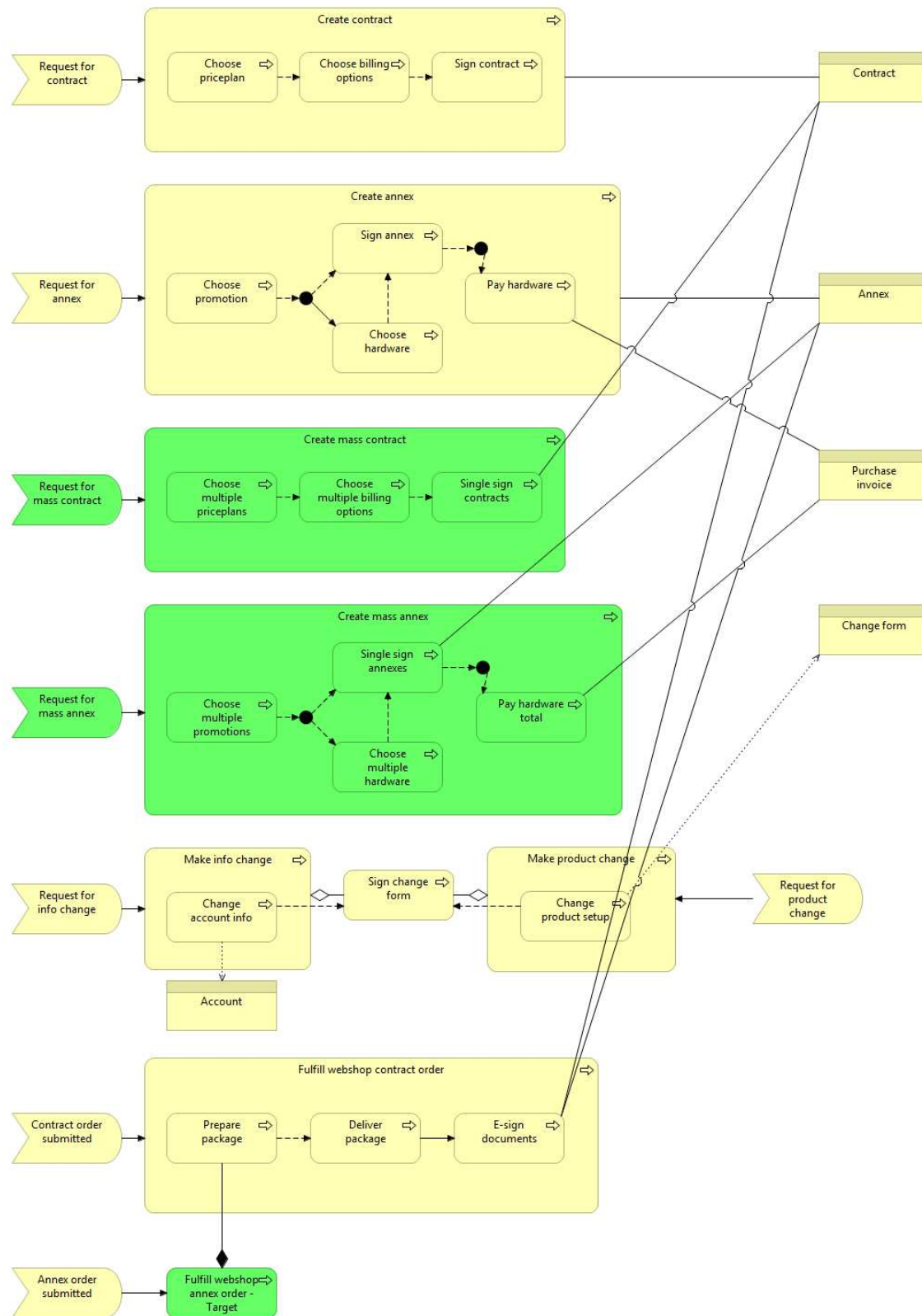
## 9.2 Ciljna poslovna arhitektura

Najpomembneje pri podpori izvajanju poslovnih procesov je, da je izvajanje popolnoma podprto. Pomembno je tudi, da je podprto brez redundance, saj redundance navadno predstavljajo sub optimalne strukture. Znotraj tega vzorca podjetja navadno poskušajo maksimirati stopnjo avtomatizacije procesnih aktivnosti in procesnega toka. Za to obstaja več razlogov kot je na primer izboljšanje časa izvajanja procesa in zmanjšanje ponavljajočih se ročnih opravil [14]. To sta eni izmed glavnih načel, ki smo jih upoštevali pri načrtovanju ciljne poslovne arhitekture in kasneje tudi ciljne informacijske arhitekture.

Ciljna poslovna arhitektura je kljub temu trenutni vsaj na prvi pogled precej podobna. Razlike so predstavljene z barvami. Rdeči gradniki v trenutni arhitekturi so procesi, ki se ukinjajo, zeleni gradniki pa so tisti, ki se v ciljni arhitekturi uvajajo. Na podlagi obeh modelov lahko naredimo analizo vrzeli med trenutno in ciljno arhitekturo.



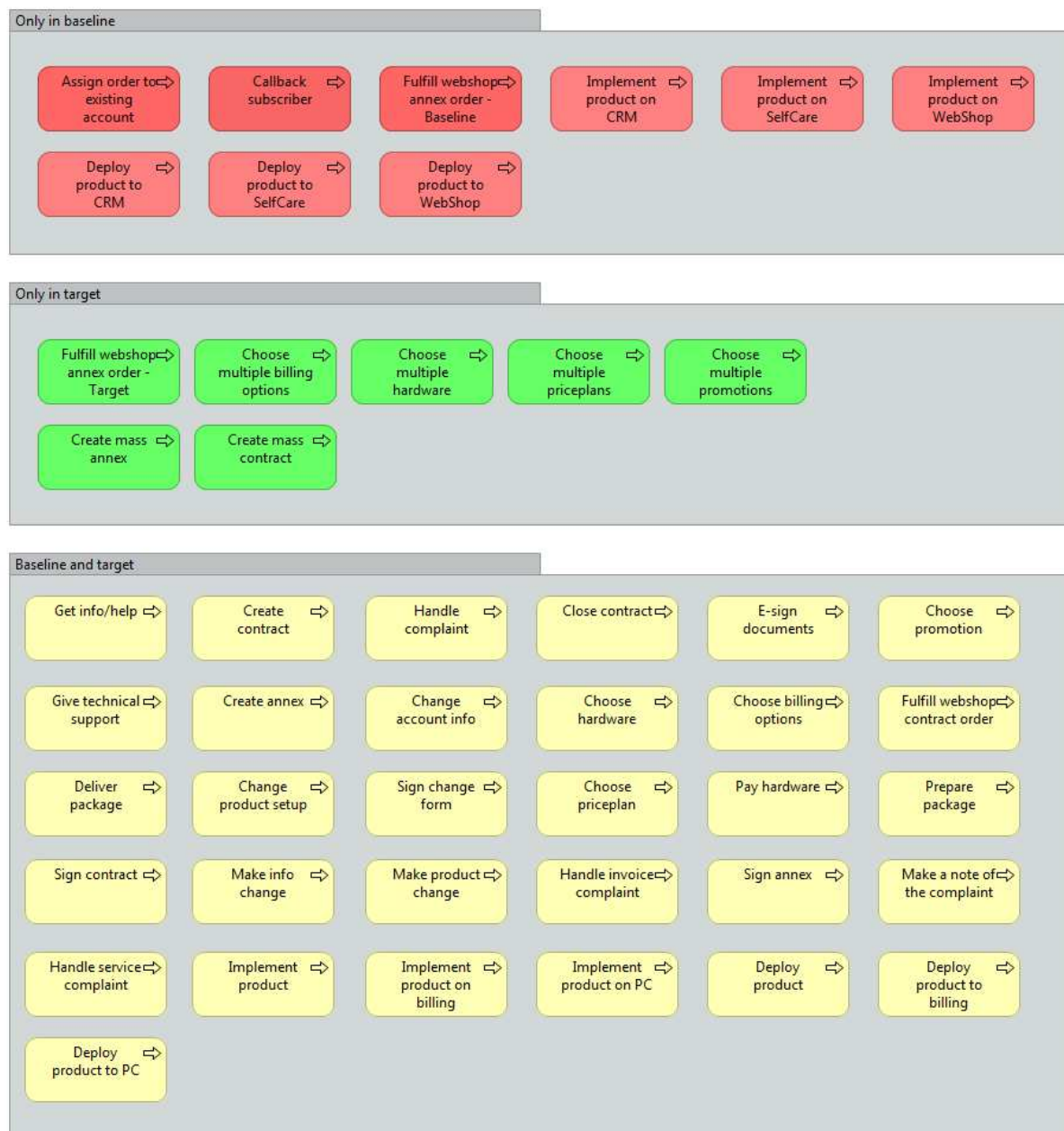
Slika 9-8: Ciljna poslovna arhitektura – 1



Slika 9-9: Ciljna poslovna arhitektura – 2



## 9.3 Analiza vrzeli poslovne arhitekture



Slika 9-10: Analiza vrzeli poslovne arhitekture

Na sliki 9-10: Analiza vrzeli poslovne arhitekture so bile povezave med procesi namenoma odstranjene za večjo preglednost. Tu se sedaj lepo vidi, katere so ključne funkcionalnosti, ki jih želimo pridobiti s posodobitvijo arhitekture, ter katere so tiste, ki bi jih želeli spremeniti oz. odstraniti.

V skupini procesov, ki jih najdemo le v trenutni arhitekturi, ne pa tudi v ciljni, najdemo izvedbo naročila aneksa preko spletne trgovine. Preostala dva procesa v skupini sta uparjanje naročila z

obstoječo stranko ter klic stranke in sta podprocesa izvedbe naročila aneksa preko spletne trgovine. V ciljni arhitekturi tega procesa ni, saj bi si želeli, da se ta proces avtomatsko izvaja, tako kot se izvaja izvedba naročila pogodbe preko spletne trgovine. V skupini procesov, ki jih najdemo le v ciljni arhitekturi je tako že omenjen avtomatizirani proces izvedbe naročila aneksa preko spletne trgovine, poleg njega pa še procesa kreiranja masovnih pogodb in masovnih aneksov ter njihovi pripadajoči podprocesi (izbira več opcij izdajanja računa in zaračunavanja, izbira različnih paketov ter izbira različnih promocij in aparatov).

Vsi preostali procesi so del tako trenutne kot ciljne arhitekture, kar pa ne pomeni nujno, da bodo v ozadju izvedeni na enak način. To je predmet faze C – Informacijska arhitektura.

## 10 Informacijska arhitektura

Po TOGAF-u se faza informacijske arhitekture deli na aplikacijsko in podatkovno arhitekturo.

Za podatkovno arhitekturo so predvideni praktično identični koraki kot za poslovno arhitekturo, vendar za pripadajočo domeno podatkov:

- Izbira referenčnih modelov, zornih kotov in orodij: kot referenčni model smo vzeli SID, zaradi omejenega nabora ne bomo uporabili posebnih zornih kotov temveč predstavili celoten pogled in sicer z diagrami iz orodja Archi [20]
- Priprava opisa trenutne podatkovne arhitekture: slika 10-1: Trenutna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti
- Priprava opisa ciljne podatkovne arhitekture: slika 10-10: Ciljna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti
- Izdelava analize vrzeli med njima: slika 10-18: Analiza vrzeli podatkovne arhitekture
- Definicija kandidatov za komponente načrta: tu se pripravi grob načrt za prehod, ki ga nato natančneje definiramo v fazi E – Implementacija arhitekture. Sprejete odločitve so tako prikazane v poglavju 12 Implementacija arhitekture.
- Razrešitev posledic na celotno arhitekturo: nova podatkovna arhitektura ne bo imela večjih posledic na hrambo in upravljanje s podatki (primer večje posledica bi bila potreba po uvedbi novih načinov dostopov ali konkretnjši tehnični transformaciji)
- Formalni pregled z deležniki
- Finalizacija podatkovne arhitekture
- Dodajanje izdelkov v dokument arhitekturne definicije

Za prikaz rezultatov teh korakov so predvideni naslednji izdelki:

- Katalog podatkovna entiteta/podatkovna komponenta
- Matrika podatkovna entiteta/poslovna funkcija
- Matrika aplikacija/podatek
- Konceptualni podatkovni diagram
- Logični podatkovni diagram

- Diagram razporeditve podatkov
- Diagram varnosti podatkov
- Diagram migracije podatkov
- Diagram življenjskega cikla podatkov

Ker v sklopu podatkovne informacijske arhitekture ne bomo predstavljali podatkovne dekompozicije do konkretnih baznih entitet, bomo večino predlaganih izdelkov izpustili. Vlogo, ki jo v izbranih procesih igrajo poslovni objekti, smo že prikazali v poslovni arhitekturi, sedaj dodajamo pregled informacijske strukture, ki pa je vrsta diagrama, ki je definirana po ArchiMate in ne po TOGAF standardu. S takim diagramom lahko predstavimo relacijo med poslovnimi in podatkovnimi objekti. Kasneje je v okviru aplikacijske arhitekture predstavljena še povezava med podatkovnimi objekti in aplikacijami. To zaokroži pregled nad razporeditvijo podatkov in podatkovnih entitet znotraj arhitekture in tako pokriva konceptualni in logični podatkovni diagram, matriko podatkovna entiteta/poslovna funkcija ter matriko aplikacija/podatek, ter lahko predstavlja tudi vsebino diagrama razporeditve podatkov (slika 10-1: Trenutna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti in slika 10-10: Ciljna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti).

Za informacijsko arhitekturo ponovimo korake, ki smo jih že spoznali pri poslovni in podatkovni arhitekturi, tokrat za informacijsko domeno:

- Izбира referenčnih modelov, zornih kotov in orodij: kot referenčni model bi kot rečeno lahko vzeli TAM, vendar se že z namenom celotnega projekta omejujemo na BSS aplikacije. Zaradi omejenega nabora ne bomo uporabili posebnih zornih kotov temveč predstavili celoten pogled in sicer z diagrami iz orodja Archi [20]
- Priprava opisa trenutne aplikacijske arhitekture: vse slike iz poglavja 10.1 Trenutna informacijska arhitektura
- Priprava opisa ciljne aplikacijske arhitekture: vse slike iz poglavja 10.2 Ciljna informacijska arhitektura
- Izdelava analize vrzeli med njima: poglavje 10.3 Analiza vrzeli informacijske arhitekture
- Definicija kandidatov za komponente načrta: tu se pripravi grob načrt za prehod, ki ga nato natančneje definiramo v fazi E – Implementacija arhitekture. Sprejete odločitve so tako prikazane v poglavju 12 Implementacija arhitekture.
- Razrešitev posledic na celotno arhitekturo: nova informacijska arhitektura ima vliv na realizacijo poslovnih procesov ter na novo tehnološko arhitekturo. Vpliv na prve je opisan v poglavju 9.2 Ciljna poslovna arhitektura in diagramih realizacije v poglavju

10.2 Ciljna informacijska arhitektura, slednji pa v poglavju 11.2 Ciljna tehnološka arhitektura.

- Formalni pregled z deležniki
- Finalizacija aplikacijske arhitekture
- Dodajanje izdelkov v dokument arhitekturne definicije

Za aplikacijsko arhitekturo so predvideni naslednji izdelki:

- Katalog aplikacijskega portfelja
- Katalog vmesnikov
- Matrika aplikacija/organizacija
- Matrika vloaga/aplikacija
- Matrika aplikacija/funkcija
- Matrika sodelovanja aplikacij
- Diagram komunikacij med aplikacijami
- Diagram lokacije aplikacij in uporabnikov
- Diagram primerov uporabe aplikacij
- Diagram možnosti upravljanja podjetja
- Diagram procesne/aplikacijske realizacije
- Diagram realizacije programske opreme
- Diagram migracije aplikacij
- Diagram razporeditve programske opreme

Izmed predlaganih izdelkov na strani aplikacijske arhitekture je za nas ključnega pomena diagram komunikacij med aplikacijami. Tekom priprave le tega nastane tudi katalog aplikacij in vmesnikov. Za povezavo s poslovno arhitekturo je pomemben tudi diagram realizacije procesov, s katerim prikažemo, katere aplikacije se uporabljajo znotraj določenih procesov. Ostale izdelke bomo na tej stopnji izpustili.

Preden pa začnemo z modeliranjem, je potrebno definirati obseg.

Začnimo s podatkovno arhitekturo. Za identifikacijo entitet si kot rečeno lahko pomagamo z ogrodjem SID. Entitete, ki nas zanimajo bomo našli v domenah stranka (Customer), produkt (Product) in trženje/prodaja (Marketing/Sales).

V strankini domeni najdemo entitete kot so stranka, interakcija s stranko, strankino naročilo, strankina težava, strankin račun ter strankino povpraševanje o računu. Ostale entitete iz te domene bomo v kontekstu te naloge zanemarili (statistike, strankina uporabljena tarifa zaračunavanja, sporazum o ravni storitev, postopek izterjave).

V produktni domeni najdemo entitete kot so produkt, produktna specifikacija, produktna ponudba in produktna konfiguracija. Ostale entitete iz te domene bomo v kontekstu te naloge zanemarili (strateški plan produktnega portfelja, produktna uspešnost, uporaba produkta, lojalnost, produktni test).

V domeni trženja in prodaje pa bomo uporabili eno samo entiteto in sicer prodajni kanal. Ostale entitete iz te domene bomo v kontekstu te naloge izpustili (plan in strategija trženja, tržni segment, tržna kampanja, konkurent, kontakt/sled/obeta, prodajne statistike).

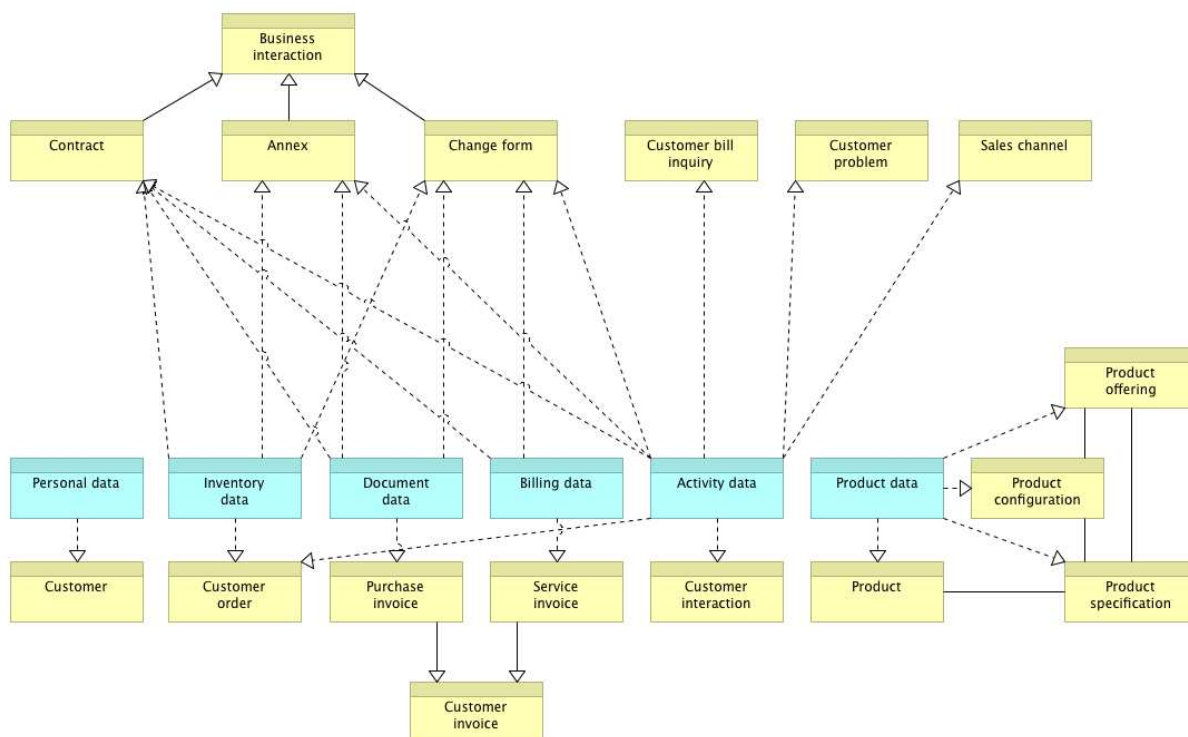
Na seznam bomo dodali še entiteto iz domene skupnih poslovnih entitet in sicer poslovno interakcijo. V našem primeru so to pogodbe, aneksi k pogodbam, obrazci za spremembe ter tudi račun za opravljene storitve in račun za izdano blago.

Entitete, ki smo jih izpustili, ne nastopajo v naših izbranih procesih, zato v tej iteraciji niso relevantne.

Na strani aplikacijske arhitekture bi lahko popis relevantnih sistemov opravili s pomočjo TAM ogrodja, vendar nam obseg aplikacijskega popisa narekuje že sama predpostavka, da spremembo arhitekture izvajamo za sisteme za podporo poslovanju. Kot smo že omenili v poglavju 2 Sistemi za podporo poslovanju v telekomunikacijah, so relevantni sistemi naslednji: sistem za obračunavanje storitev, sistem za upravljanje z naročili, sistem za upravljanje odnosov s strankami, sistem za upravljanje klicnega centra ter vse, kar je lahko del enega izmed naštetih sistemov ali pa je neposredno tudi vsebinsko povezan z njimi.

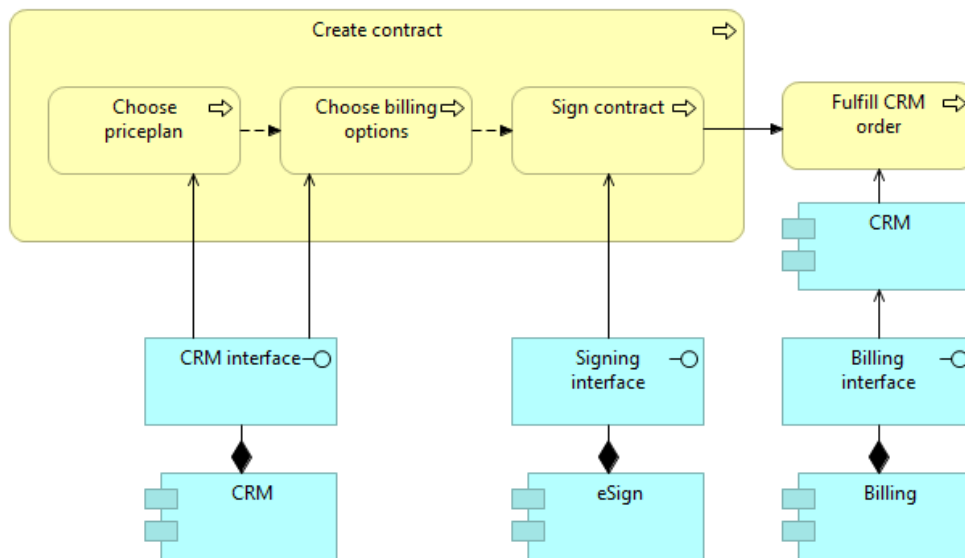
## 10.1 Trenutna informacijska arhitektura

S podatkovnega vidika je arhitektura relativno preprosta. Izbrane entitete se prepletajoče hranijo v podatkovne objekte, ki vsebujejo osebne podatke, inventar, dokumente same, podatke o obračunu in podatke o aktivnostih stranke in sicer tako, kot je prikazano na sliki 10-1: Trenutna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti. Izbrane entitete so predstavljene v obliki poslovnih objektov, ki so sicer predmet poslovne arhitekture. Kako so uporabljene smo videli že v poglavju 9 Poslovna arhitektura, njihovo konkretno obliko pa predstavljajo podatkovni objekti ali fizične reprezentacije.



Slika 10-1: Trenutna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti

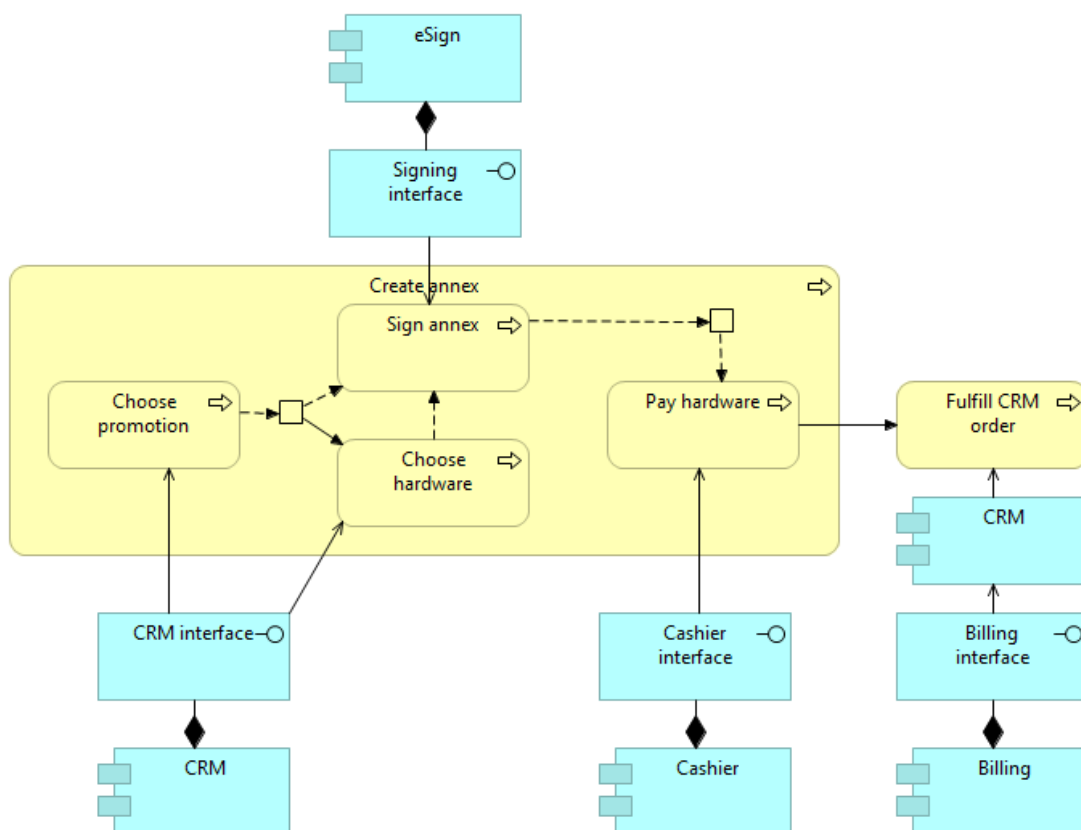
Čeprav je lastnik večine teh objektov obračunski sistem, se trenutna informacijska arhitektura nasploh izjemno zanaša na CRM, ki vsebinsko skrbi tako za splošno obravnavo strank kot za prodajo in pritožbe ter informacije. S pomočjo diagramov procesne/aplikacijske realizacije lahko natančno predstavimo, v katerih primerih se uporablja.



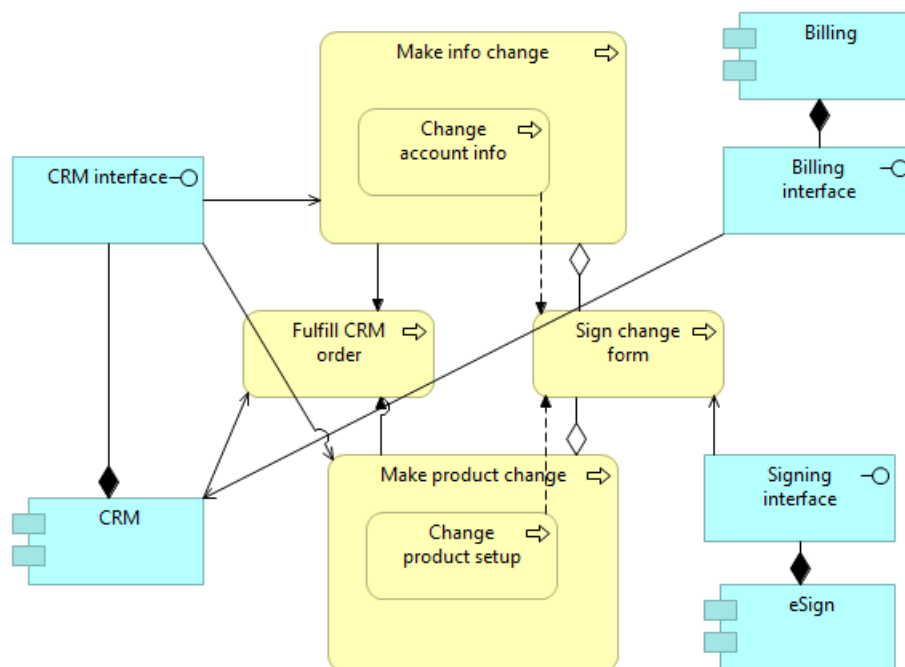
Slika 10-2: Diagram trenutne realizacije procesa priprave pogodbe

Za proces priprave pogodbe se CRM uporablja v praktično vseh korakih. Poskrbi tako za izbiro produktov kot za izvedbo storitve. Enako velja za proces priprave aneksa na pogodbo. Ta je skoraj identičen, vendar je zaradi dodatnega koraka plačila aparata udeležen tudi blagajniški sistem. Pri obeh najdemo tudi sistem za podpisovanje dokumentov, ki je nasploh prisoten pri vseh procesih, ki zahtevajo podpis določenega obrazca s strani stranke (slika 10-2: Diagram trenutne realizacije procesa priprave pogodbe in slika 10-3: Diagram trenutne realizacije procesa priprave aneksa k pogodbi).



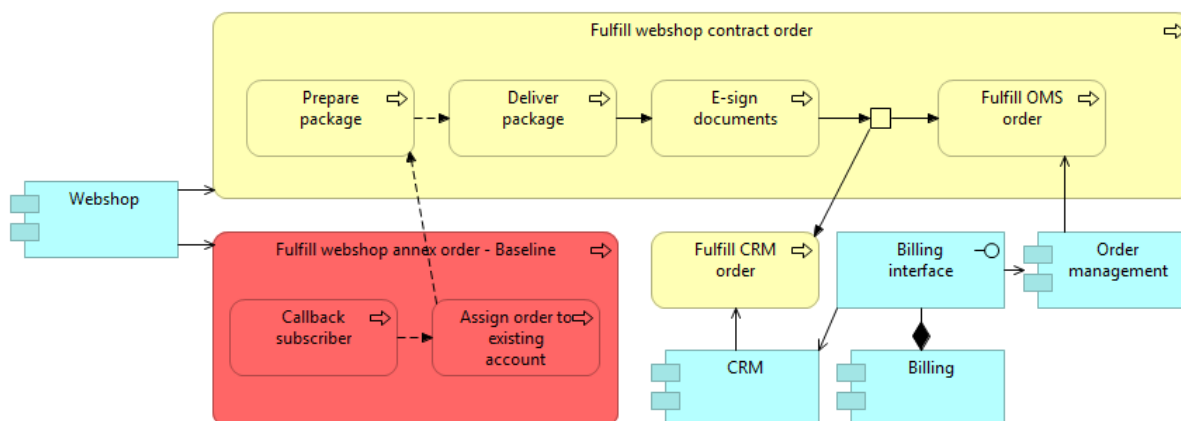


Slika 10-3: Diagram trenutne realizacije procesa priprave aneksa k pogodbi



Slika 10-4: Diagram trenutne realizacije procesov izvajanja sprememb

Tudi pri izvajanju sprememb bodisi produktnih (npr. dodajanje nove storitve) ali informacijskih (npr. sprememba naslova stranke) je izvedba popolnoma na strani CRM sistema. Ta poleg vmesnika, ki te spremembe omogoča, prevzema tudi funkcijo izvajanja storitve in sicer preko vmesnika obračunskega sistema.

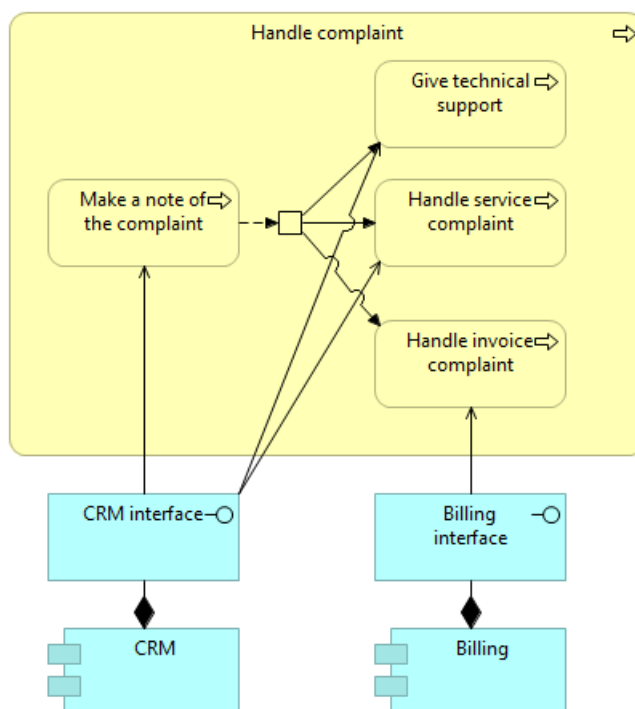


Slika 10-5: Diagram trenutne realizacije procesov za pripravo pogodb in aneksov preko spletne trgovine

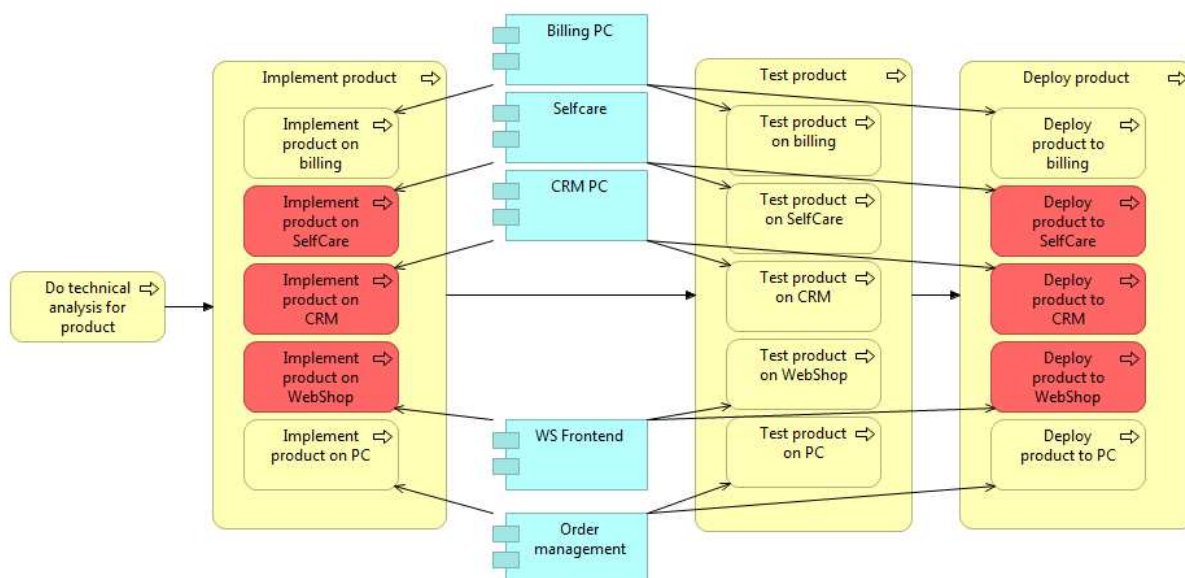
V primeru naročil preko spleta je postopek drugačen. Za izvedbo celotnih procesov skrbita spletna trgovina sama ter OMS. Edina izjema je primer, ko gre za izvedbo aneksa na pogodbo za obstoječega naročnika. V tem primeru zaključka in izvedbe naročila ne izvedeta spletna trgovina in OMS temveč je naročilo preusmerjeno na CRM in tam tudi dokončano in poslano v izvedbo (slika 10-5: Diagram trenutne realizacije procesov za pripravo pogodb in aneksov preko spletne trgovine).

Ravnanje s težavami je proces, ki se izvaja v klicnem centru. Tudi tam je primarno orodje, ki se ga vsakodnevno uporablja, CRM. Poenostavljeno smo težave razdelili v tri kategorije: tehnične težave, težave s storitvami ter težave z računi.

Tehnične težave smo poimenovali tiste, ki niso posledica napačnega delovanja omrežja ali ostalih sistemov ponudnika, temveč posledica napačnih nastavitvev ali neznanja uporabnika. Za reševanje tovrstnih težav agenti klicnega centra uporabljajo bazo znanja, ki je del CRM-ja in s pomočjo katere uporabnikom svetujejo pri odpravljanju tovrstnih težav. Težave s storitvami so tiste, ki so posledica bodisi napak pri delovanju omrežja ali ostalih sistemov ponudnika. Če je težava posledica slednjih, je potrebno bodisi popraviti nastavitve storitev, jih ponovno aktivirati ali popolnoma spremeniti. Vse te akcije se bodisi neposredno ali posredno, kot prijavo napake v zaledne sisteme, reši preko vmesnika CRM sistema. V primeru omrežnih težav se to z vidika klicnega centra ponovno rešuje preko CRM-ja in sicer z zabeleženjem napake in prijavo sistemom na OSS strani za obravnavo (slika 10-6: Diagram trenutne realizacije procesa za ravnanje s pritožbami). Težave z računi se rešujejo neposredno v obračunskem sistemu.



Slika 10-6: Diagram trenutne realizacije procesa za ravnanje s pritožbami

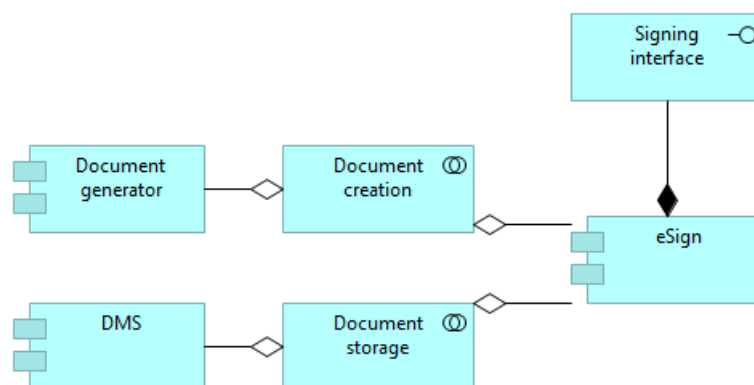


Slika 10-7: Diagram trenutne realizacije procesa za implementacijo novega produkta

Proces implementacije novega produkta zajema tehnično analizo, samo implementacijo produkta, testiranje ter prenos sprememb na produkcijske sisteme. Prvi korak ni sistemsko realiziran, vendar kljub temu zahteva resurse z vseh vpletenih sistemov. Sledijo implementacija, testiranje in prenos in vsi trije koraki zahtevajo delo na vseh vpletenih sistemih. To so: obračunski sistem, selfcare portal, CRM, spletna trgovina in produkti katalog.

Testiranje produkta kot celote se izvaja na vseh udeleženi sistemih (slika 10-7: Diagram trenutne realizacije procesa za implementacijo novega produkta).

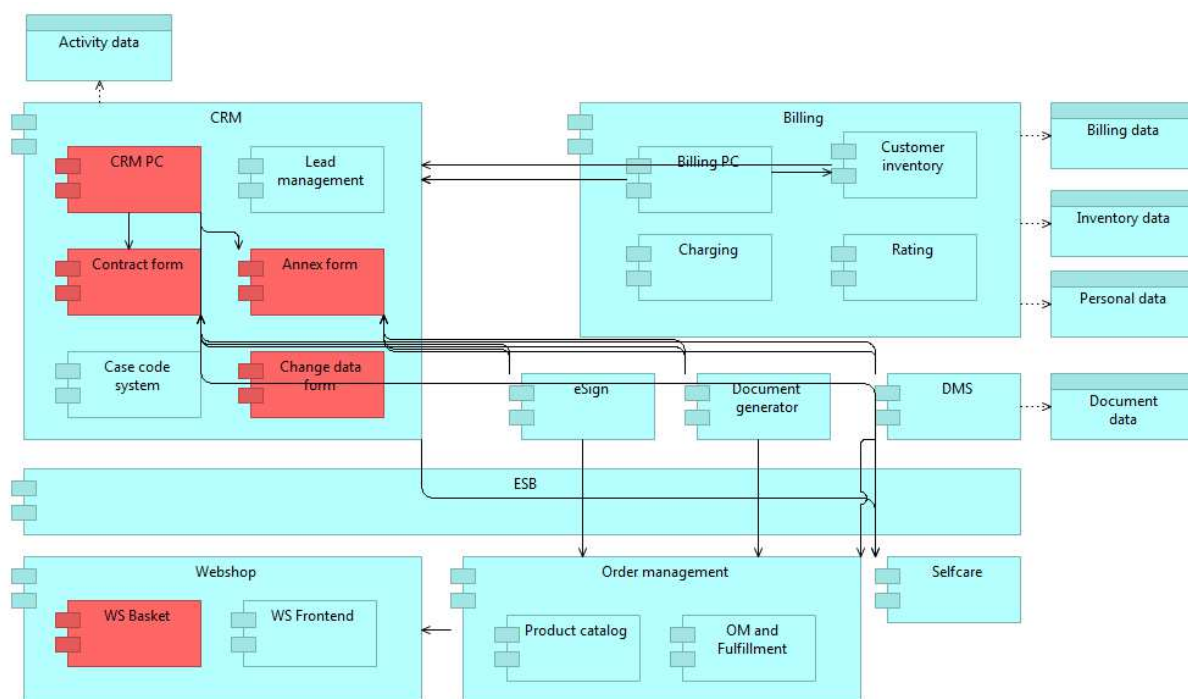
Na tem mestu naj dodamo, da je z namenom poenostavitve modela povsod izpuščeno kreiranje in shranjevanje dokumentov. Na prikazanih diagramih realizacije ter prav tako na diagramih procesnih tokov iz poglavja 9 Poslovna arhitektura je prikazano le podpisovanje dokumentov, v resnici pa je potrebno dokumente najprej kreirati, nato podpisati in na koncu še shraniti. Realen diagram sodelovanja med aplikacijami je zato tak, kakršen je prikazan na sliki 10-8: Diagram sodelovanja aplikacij za podpisovanje dokumentov.



Slika 10-8: Diagram sodelovanja aplikacij za podpisovanje dokumentov

Nadaljujemo s pregledom sodelovanja med ostalimi aplikacijami, kar je ključen pogled v domeni aplikacijske informacijske arhitekture.

Ob pregledu diagramov realizacije ugotovimo, da sta ključna sistema, ki se najpogosteje pojavljata na njih, obračunski sistem ter CRM. Komunikacija med njima večinoma poteka direktno, vsi preostali sistemi pa med seboj komunicirajo preko storitvenega vodila (enterprise service bus). Te povezave so izpuščene in narisane direktno med sistemi, da je model preglednejši ter da je vidno, kateri sistemi med seboj dejansko komunicirajo.



Slika 10-9: Trenutna aplikacijska arhitektura

Ta arhitektura ima štiri ključne pomanjkljivosti:

- Produktni katalog je definiran na več lokacijah (obračunski sistem, CRM, OM)
- Dostava storitev je izvedena dvakrat (CRM, OM)
- Povezava med dvema ključnima sistemoma je direktna (CRM - obračunski sistem)
- Nakupi in spremembe so možni na treh različnih kanalih, ki pa niso povezani (spletna trgovina, selfcare portal, prodajna mesta)

### ***Produktni katalog je definiran na več lokacijah***

CRM, obračunski sistem in OM imajo vsak svoj produktni katalog, med katerimi ni ne sinhronizacije, ne preslikave. To pomeni, da je razvoj novih produktov dolgotrajen, prav tako spreminjanje obstoječih, vzdrževanje in odpravljanje napak. Sinhronizacija med njimi ne obstaja in bi tudi sicer bila tehnično izjemno zahtevna, saj obstaja preslikava med CRM in obračunskim produktnim katalogom, ter med OM in obračunskim produktnim katalogom, ne pa med CRM in OM produktnim katalogom. Posledično je lista aktivnosti strank pomanjkljiva v primeru, ko se le-te izvajajo preko različnih kanalov. Prav tako je izpeljava ene aktivnosti preko dveh različnih kanalov nemogoča. Kot je že bilo opisano v poglavju 9 Poslovna arhitektura, sicer obstaja poslovni proces, ki omogoča, da se nakup, ki se začne na spletni strani, zaključi s pomočjo agenta po telefonu, vendar to vključuje ročno delo agenta, ki produkte iz spletne košarice prepozna po opisu in izbere njihovo ekvivalentno konfiguracijo.

*Dostava storitev je izvedena dvakrat*

Težava je podobna kot v primeru več produktnih katalogov. Dostava storitev je predmet tako CRM kot OM sistema, kar pomeni dvojni razvoj in dvojno vzdrževanje, prav tako pa tudi dvojno integracijo s sistemi, ki so v ta proces vpleteni.

*Povezava med dvema ključnima sistemoma je direktna*

Kot rečeno velika večina komunikacije med BSS sistemi poteka preko storitvenega vodila. To omogoča enotno sledljivost in beleženje ter šibko sklopljenost med sistemi. Po drugi strani pa glavčina komunikacije med dvema ključnima sistemoma poteka direktno, kar pomeni, da se velika količina komunikacije beleži v svoj sistem in izvaja tesno sklopljeno, zaradi česar je vzdrževanje in odpravljanje napak oteženo, potencialne večje spremembe ali zamenjava teh dveh sistemov pa je otežena.

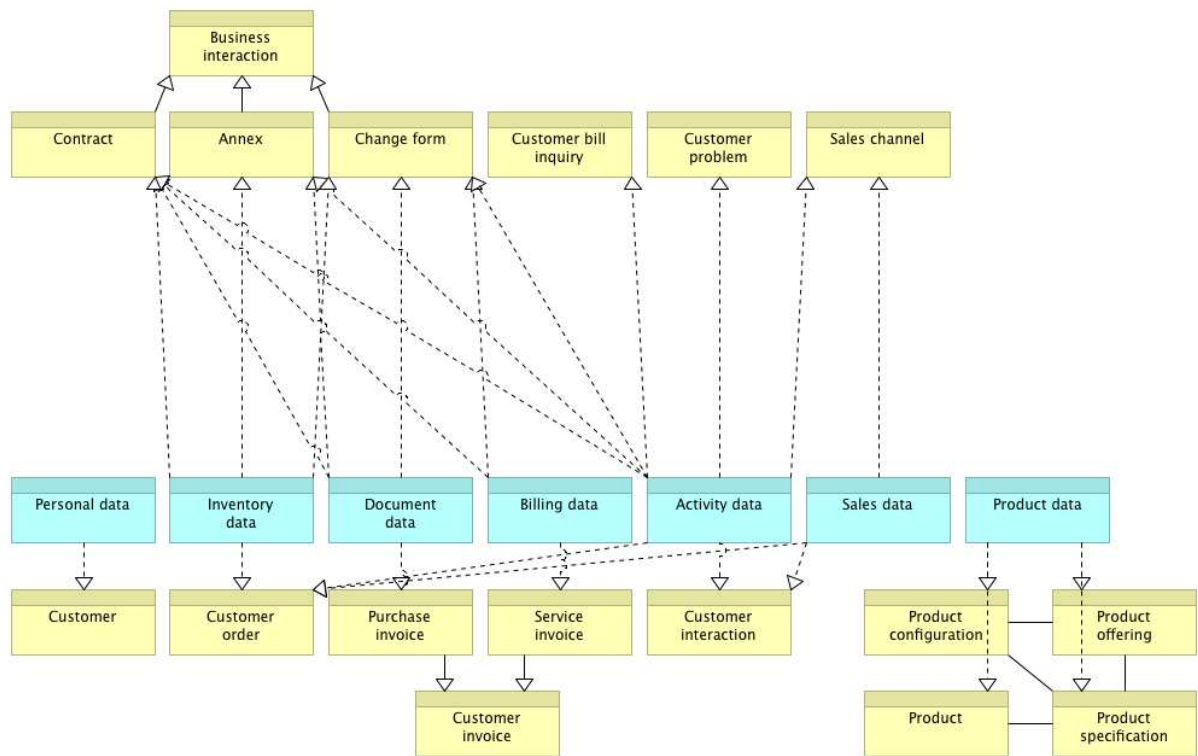
*Nakupi in spremembe so možni na treh različnih kanalih, ki pa niso povezani*

Kombinirana prodaja preko več kanalov je eden izmed ciljev, ki so bili definirani v preliminarni fazi in arhitekturni viziji in kot taka predstavlja eno ključnih pomanjkljivosti trenutne arhitekture. Želja torej je, da bi se dalo nakup začeti preko enega kanala in zaključiti preko drugega z jasnim pregledom dogajanja. V trenutni postavitvi to onemogočata prvi dve omenjeni pomanjkljivosti, torej večkratni produktni katalog in dostava storitev.

## 10.2 Ciljna informacijska arhitektura

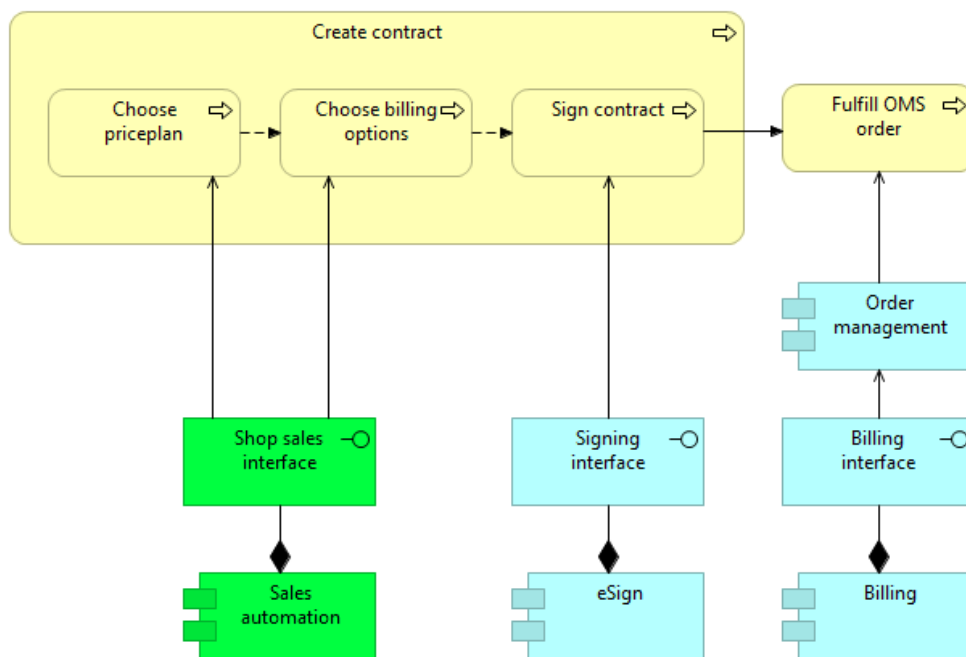
Iz podatkovnega vidika imamo med obstoječo in ciljno arhitekturo eno samo razliko in sicer vpeljavo eksplicitnih prodajnih podatkov. V trenutni podatkovni arhitekturi so namreč bili predstavljeni le v obliki pogodb in naročil, sedaj pa to ločujemo. Razlog za ločevanje je vpeljava ločene prodajne aplikacije (slika 10-10: Ciljna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti).

Sicer pa razmerja med poslovnimi in podatkovnimi objekti ostajajo enaka, spremeni se le lastništvo nad podatkovnimi objekti v smislu njihove relacije do aplikacij, kar pa je predmet aplikacijske arhitekture in bo tudi podrobneje predstavljeno v okviru le-te.



Slika 10-10: Ciljna podatkovna arhitektura - povezava med poslovnimi in podatkovnimi objekti

V izogib identificiranim pomanjkljivostim trenutne arhitekture je v ciljni aplikacijski arhitekturi predvidenih kar nekaj sprememb.

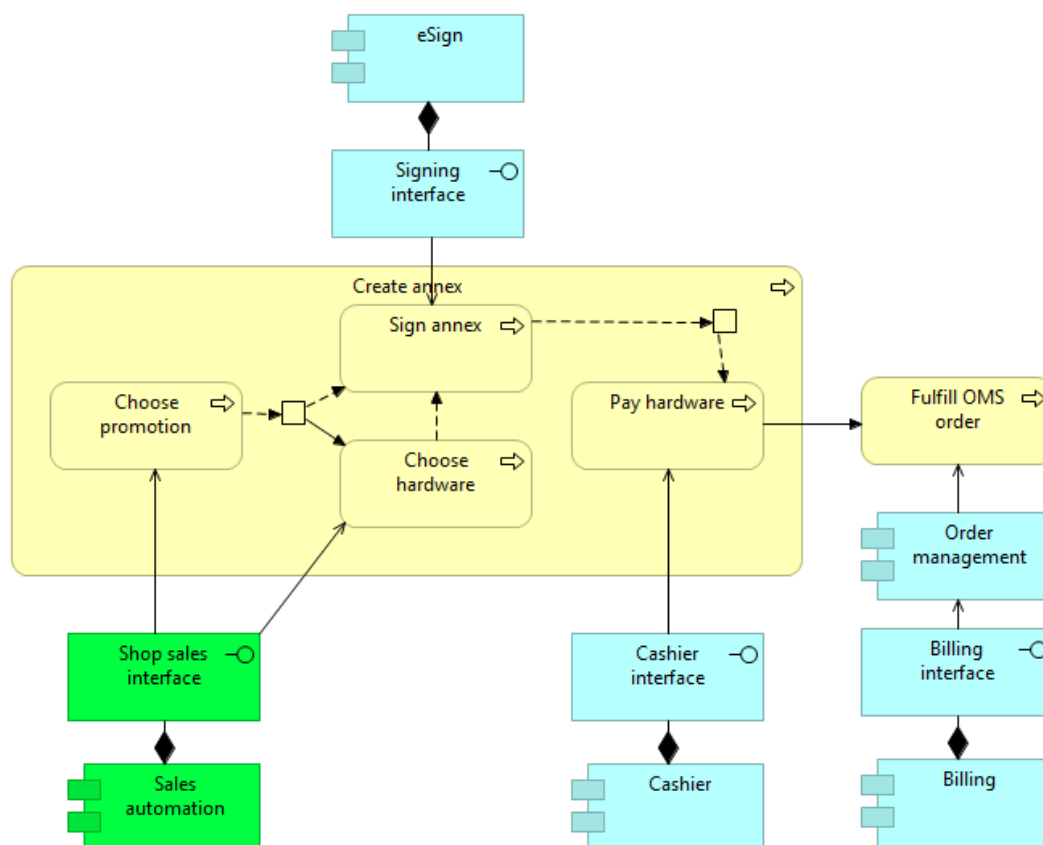


Slika 10-11: Diagram ciljne realizacije procesa priprave pogodbe

Ciljni proces priprave pogodbe je že bistveno drugačen od trenutnega. Sam proces sicer poteka enako, vendar je realiziran na popolnoma drugačen način. Prvi koraki, ki predstavljajo prodajni del, se izvajajo s pomočjo novega prodajnega sistema. Po podpisu pogodbe, se naročilo izvede preko OMS in ne direktno iz sistema, ki je naročilo pobral (slika 10-11: Diagram ciljne realizacije procesa priprave pogodbe).

Enako velja za proces priprave aneksa k pogodbi (slika 10-12: Diagram ciljne realizacije procesa priprave aneksa na pogodbo).

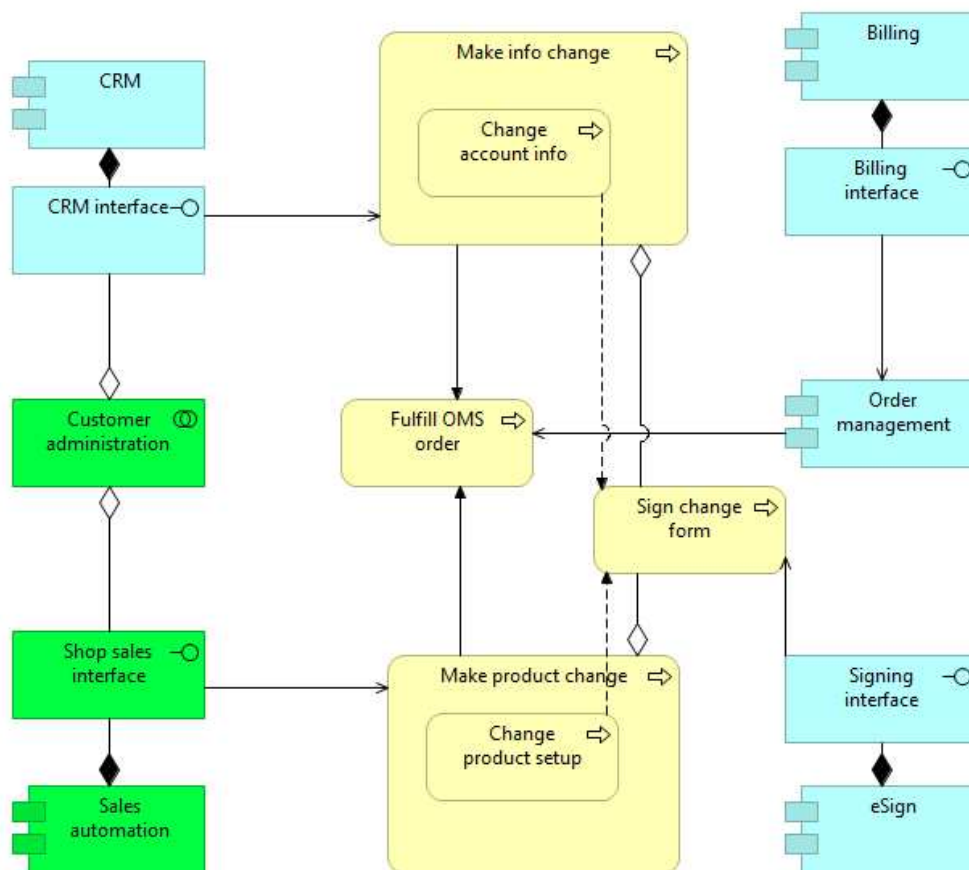




Slika 10-12: Diagram ciljne realizacije procesa priprave aneksa na pogodbo

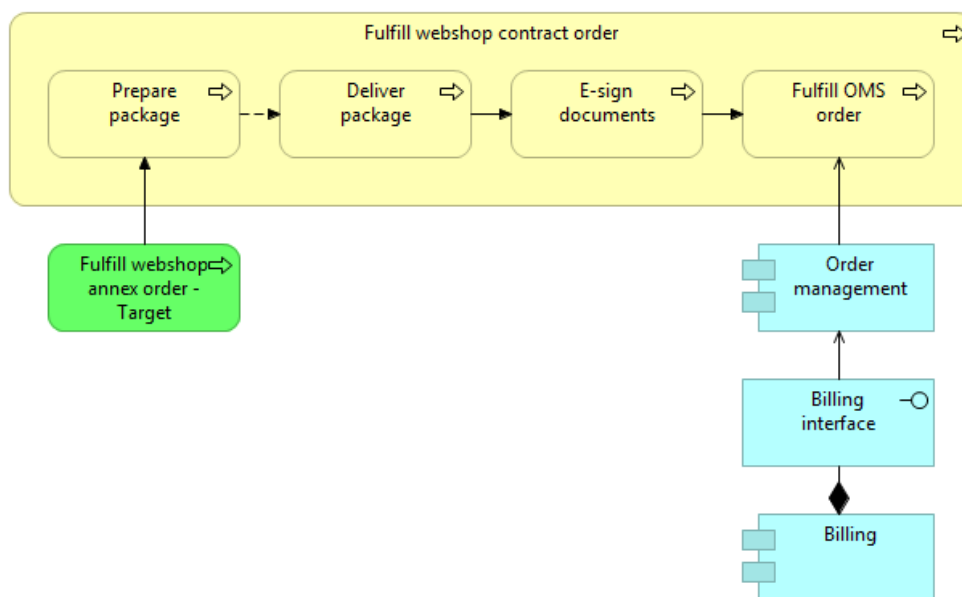
Pri pregledu procesa za izvajanje sprememb je lepo vidno končno stanje in porazdelitev odgovornosti med prodajnim sistemom in CRM sistemom. Prvi je osredotočen na storitve in njihovo prodajo, slednji pa je osredotočen na stranko in njene attribute. Oba sistema sta združena v isto masko, kar uporabnikom omogoča bolj tekoče delo. To je eksplicitno prikazano z elementom sodelovanja med obema sistemoma.

Razlika v primerjavi s trenutno arhitekturo je poleg uvedbe posebnega prodajnega orodja tudi v premiku izvedbe naročil iz CRM sistema na OMS kot smo že videli pri procesih za pripravo pogodbe in aneksa na pogodbo (slika 10-13: Diagram ciljne realizacije procesov izvajanja sprememb).

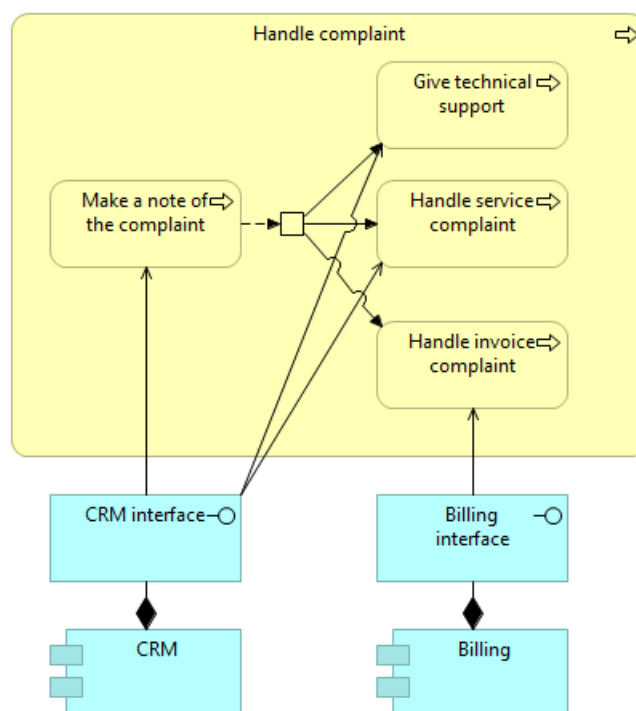


Slika 10-13: Diagram ciljne realizacije procesov izvajanja sprememb

Pri procesih priprave pogodbe in aneksa na pogodbo preko spletne trgovine je v ciljni arhitekturi ena sama, a zelo pomembna sprememba: aneksi za obstoječe naročnike se ne izvajajo več s posredovanjem CRM sistema, temveč se v celoti izvedejo po enakem procesu kot se pripravljata pogodba ter aneks za nove naročnike. Tako ni več potrebno posredovanje dodatnega agenta, izvedba naročila pa se izvede preko OMS (slika 10-14: Diagram ciljne realizacije procesov za pripravo pogodb in aneksov preko spletne trgovine).

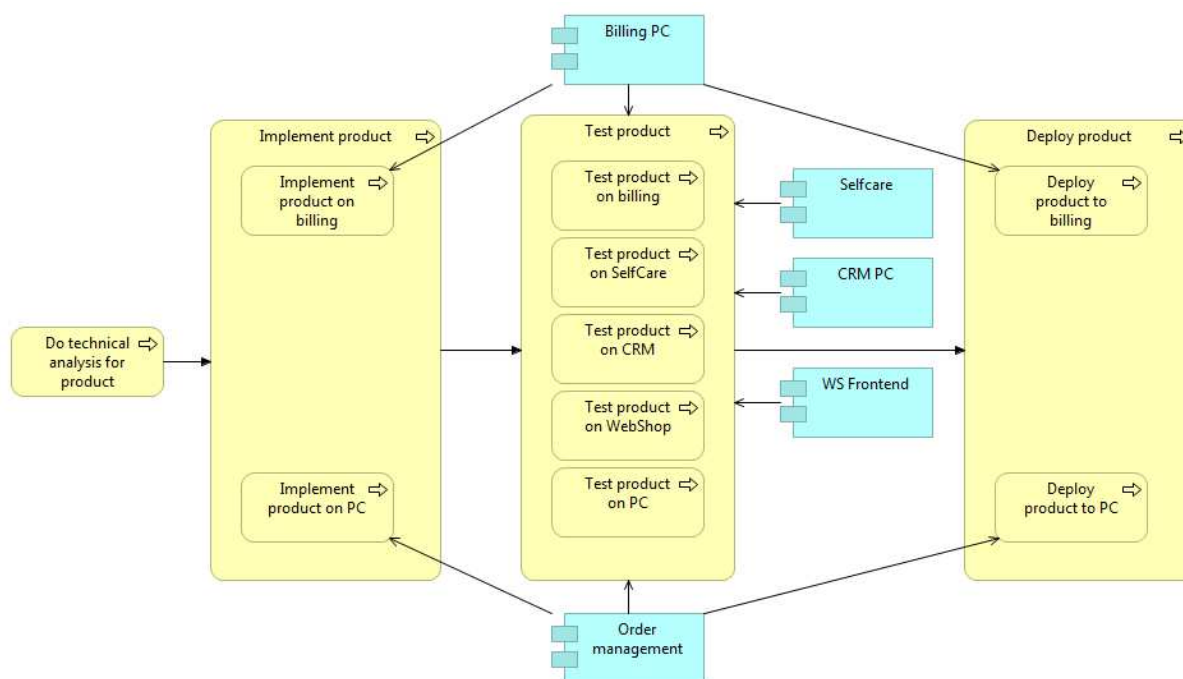


Slika 10-14: Diagram ciljne realizacije procesov za pripravo pogodb in aneksov preko spletne trgovine



Slika 10-15: Diagram ciljne realizacije procesa za ravnanje s pritožbami

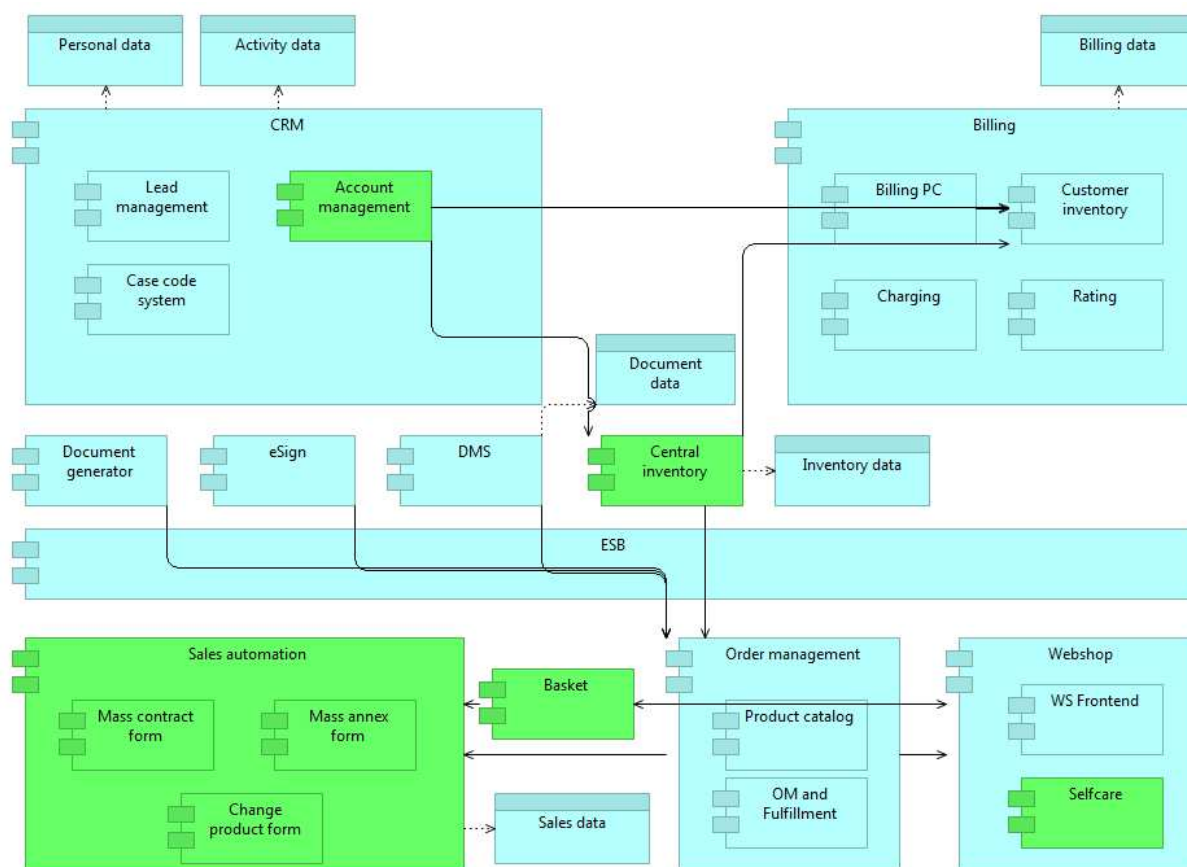
Proces ravnanja s pritožbami je edini obravnavani proces, katerega realizacija (in tudi proces sam) se v ciljni arhitekturi popolnoma nič ne spremeni (slika 10-15: Diagram ciljne realizacije procesa za ravnanje s pritožbami). Sprememb ni, ker sam proces ravnanja s pritožbami ni problematičen, tudi nikakor ne vpliva na definirane cilje. Na cilje vpliva sama pojavljanje težav, kar pa se rešuje drugje.



Slika 10-16: Diagram ciljne realizacije procesa za implementacijo novega produkta

Proces implementacije novega produkta se drastično spreminja. Tehnična analiza ostaja nespremenjena, saj je ne realizira noben sistem. Implementacija produkta ni več potrebna na vseh sistemih, temveč le na obračunskem sistemu in produktnem katalogu, kar prinaša okrnjen proces in manjšo udeležbo sistemov, potrebnih za njegovo realizacijo. Enako velja za prenos sprememb na produkcijo. Testiranje zaradi posrednega vpliva na ostale sisteme ostaja enako kot na trenutni arhitekturi in zahteva delo na vseh udeleženi sistemih (slika 10-16: Diagram ciljne realizacije procesa za implementacijo novega produkta).

S pregledom sodelovanja z drugačne perspektive vidimo že identificirane spremembe (slika 10-17: Ciljna aplikacijska arhitektura).



Slika 10-17: Ciljna aplikacijska arhitektura

### *Account management*

Stari CRM nima komponente za celostno obravnavo stranke. Vpis njenih začetnih podatkov se zgodi v okviru sklepanja prve pogodbe, sami podatki so shranjeni v obračunskem sistemu, aktivnosti stranke pa so predmet sistema kodnih primerov (case code system), ki pa ni celostno integriran v enoten pregled stranke. Z vpeljavo te komponente se omenjeni koncepti centralizirajo.

### *Central inventory*

Centralni inventar je komponenta, ki drži stanje in zgodovino storitev ter ostalih produktov, ki jih stranka ima. Inventar v obračunskem sistemu namreč ne vsebuje komercialnih produktov, saj je to na nek način že zaledni sistem, katerega vsebina je namenjena lastni uporabi in ne preostalim sistemom. Prav tako ne vsebuje celostne zgodovine. Naj podam primer: promocija, ki vsebuje popust na določeno storitev, je v obračunskem sistemu ponazorjena kot storitev in popust nanjo, v centralnem inventarju pa je ponazorjena v obliki promocijskega produkta. Ta komponenta torej vsem kanalom omogoča, da imajo enoten in jasen pregled nad inventarjem stranke. Brez podatka iz inventarja bi lahko kombinacijo produkta in popusta lahko interpretirali na več načinov.

*Mass contract form in Mass annex form*

Obrazca za masovno sklepanje pogodb in aneksov sta del novega prodajnega sistema in hkrati pokrivata tudi sklepanje posameznih pogodb in aneksov. Z njima je zadoščeno zahtevam ciljne poslovne arhitekture, ki predvideva procese sklepanja tovrstnih pogodb.

*Change product form*

Obrazec za spremembo produktov je še ena komponenta novega prodajnega sistema. Zamenjuje staro CRM komponento Obrazec za spremembe, ki je vključevala tako spremembe osebnih in obračunskih podatkov kot spremembo produktov. Ker spremembe osebnih in obračunskih podatkov pokriva komponenta za upravljanje s strankami, ob zamenjavi manjka le še funkcionalnost spremembe produktov. Za ta obrazec ni predvidena različica za masovno spreminjanje, ker gre za bolj specifične spremembe, ki se le redko delajo na večjem številu strank.

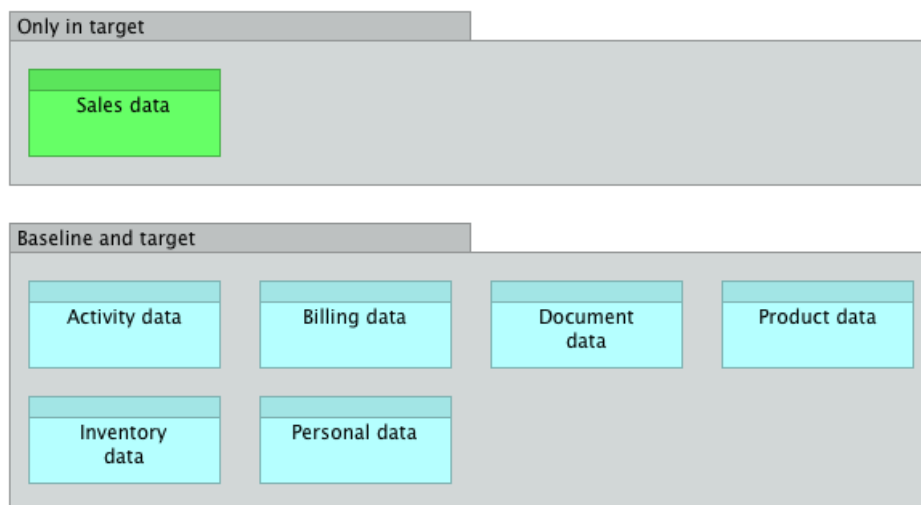
*Basket*

Košarica zamenjuje staro komponento košarice spletne trgovine. Novo košarico uporabljata tako spletna trgovina kot prodajni sistem in predstavlja integracijsko točko med njima. To je ena izmed ključnih komponent pri zagotavljanju cilja po izvedbi prodajnega procesa skozi več kanalov.

*Selfcare*

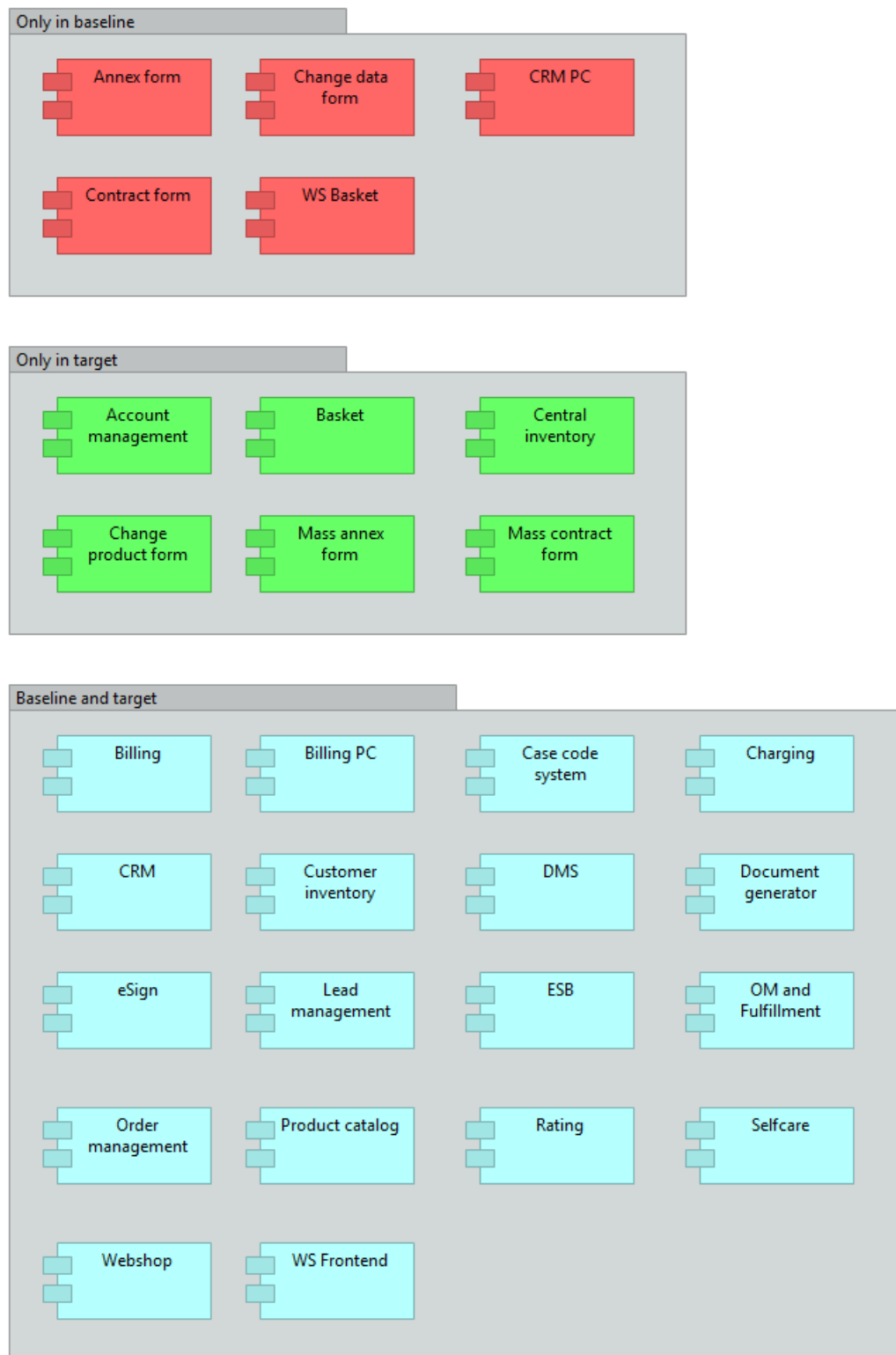
Selfcare portal načeloma ni nova komponenta, vendar je na modelu označena z zeleno in jo v tem poglavju obravnavamo zaradi njene spremenjene postavitve. V stari arhitekturi je namreč to bila samostojna komponenta, ki je bila tesno povezana s CRM sistemom. V ciljni arhitekturi je to integriran del spletne trgovine in je navzven nevidno vpletena vanj. Namen je brisanje meje med nakupnim procesom, kakršnega poznamo iz stare spletne trgovine in spreminjanje produktov obstoječih strank, ki ga sicer obravnava portal.

## 10.3 Analiza vrzeli informacijske arhitekture



*Slika 10-18: Analiza vrzeli podatkovne arhitekture*

Kot smo že omenili, je edina razlika med trenutno in ciljno podatkovno informacijsko arhitekturo vpeljava podatkovnega objekta »Sales data« (slika 10-18: Analiza vrzeli podatkovne arhitekture). Večino sprememb nosi aplikacijska informacijska arhitektura kot vidimo na sliki 10-19: Analiza vrzeli aplikacijske arhitekture.



Slika 10-19: Analiza vrzeli aplikacijske arhitekture

Poglavitne razlike med trenutno in ciljno aplikacijsko arhitekturo so že bile predstavljene v poglavjih 10.1 in 10.2, slika 10-19: Analiza vrzeli aplikacijske arhitekture pa prikazuje model analize vrzeli in služi kot hiter pregled komponent, ki se ohranjajo, vpeljujejo in ukinjajo.



Naj na tem mestu poudarim, da ne gre le za odstranjevanje in dodajanje komponent. Nekatere komponente, ki se ohranjajo, bodo zaradi zamenjave drugih ključnih komponent potrebovale obsežno prenovo.

Poglejmo, kakšen je vpliv na vsako izmed njih:

- Billing: obračunski sistem kot celota lahko ostane, kakršen je
- Billing PC: produktne katalogu v okviru obračunskega sistema se je praktično nemogoče izogniti, saj le-ta drži podatke o načinu obračunavanja teh produktov; tukaj ni sprememb
- Case code system: princip sistema za obravnavanje interakcij s strankami ostane enak, urediti je treba njegovo integracijo s komponento za upravljanje s strankami (Account management)
- Charging: tudi ta komponenta obračunskega sistema ne potrebuje sprememb
- CRM: kot celota ne potrebuje posebnih sprememb, le-te se dogajajo na njegovih posameznih komponentah
- Customer inventory: zgodba je tu podobna kot s produktnim katalogom obračunskega sistema – ni se mu mogoče izogniti, saj le preko njega obračunski sistem ve, kaj mora določeni stranki zaračunati; ni sprememb
- DMS: ta komponenta ne potrebuje nobene spremembe
- Document generator: komponenta sicer ne potrebuje konceptualnih ali integracijskih sprememb, vendar je zaradi sprememb v prodajnih procesih potrebno spremeniti tudi določene obrazce, ki jih ta komponenta generira, gre torej le za vsebinske spremembe (npr. masovne pogodbe)
- eSign: ta komponenta ne potrebuje nobene spremembe
- Lead management: potrebna je integracija s komponento za upravljanje s strankami (account management), da je možno priložnost spremeniti v stranko in ohranjati povezavo med njima
- ESB: ker komunikacija med komponentami poteka preko storitvenega vodila, je le to potrebno prilagoditi povsod, kjer se vsaj ena izmed komponent v pogovoru spremeni. Dodatno je potrebno implementirati povezave med CRM in obračunskim sistemom, saj ta komunikacija v trenutni arhitekturi poteka direktno, čemur bi se v ciljni arhitekturi želeli izogniti. Storitveno vodilo nosi največje posredne spremembe izmed vseh komponent. Ocenjujemo, da se lahko spremeni celo do 80% vseh njegovih aplikacij.
- OM and fulfillment: sistem za upravljanje z naročili in za dostavljanje storitev je treba nadgraditi s sposobnostjo dostavljanja storitev obstoječim strankam, z odstranjevanjem storitev ter generalno z dostavljanjem določenih produktov, ki zaradi omejitev OM kot celote v trenutni arhitekturi niso potrebni (glej spodaj “Product catalog”)

- Order management: dodati mu je potrebno integracijo s centralnim inventarjem, ostale spremembe so predmet njegovih komponent.
- Product catalog: produkti katalog v trenutni arhitekturi uporabljamo le ob naročilih za povsem nove stranke, kar pomeni, da mu manjkajo določeni produkti, ki niso več na voljo, vendar jih stare stranke še uporabljajo, migracijska pravila med različnimi storitvami ter ugodnosti in posebnosti, ki jih imajo obstoječe stranke. Ta vsebinski primanjkljaj je precej obširen, saj je za nadgradnjo potrebno zajeti veliko število poslovnih pravil.
- Rating: ta komponenta ne potrebuje sprememb
- Selfcare: velik posredni vpliv na selfcare portal smo že omenjali, saj se komponenta seli v spletno trgovino in postaja njegov integriran del. V trenutni aplikacijski arhitekturi je selfcare portal tesno sklopljen s CRM sistemom, v ciljni pa mora uporabljati produkti katalog in centralni inventar. Tudi z vidika predstavitvenega nivoja so spremembe velike, saj ciljna arhitektura predvideva njegovo nevidno integracijo z uporabniškim vmesnikom spletne trgovine.
- Webshop: za podporo obstoječim strankam v ciljni arhitekturi spletna trgovina potrebuje registracijo uporabnikov in njihovo identifikacijo. Ostale spremembe so na nivoju njenih komponent.
- WS Frontend: spremembe na uporabniškem vmesniku so potrebne zaradi integracije selfcare portala ter zaradi zamenjave košarice (košarica spletne trgovine se namreč zamenjuje z generalno košarico, namenjeno vsem prodajnim kanalom).

## 11 Tehnološka arhitektura

Faza D – Tehnološka arhitektura je tretja faza dejanske opredelitve arhitekture same. Tekom nje je potrebno pripraviti trenutno in ciljno tehnološko arhitekturo ter analizo vrzeli med njima. Za modeliranje vseh treh uporabimo jedro ArchiMate ogrodja in sicer vse tri strukture na tehnološki ravni.

Popis vseh vozlišč, povezav, sistemske programske opreme ter ostalih konceptov tehnološke arhitekture je lahko zelo širok. Ker glede na namen celotnega cikla poslovno-informacijske arhitekture, ki ga obravnavamo, ne predvidevamo bistvenih sprememb na tehničnem delu arhitekture, se bomo omejili na prikaz posledic, ki jih imajo spremembe informacijske arhitekture na tehnološko arhitekturo.

Koraki faze tehnološke arhitekture so ponovno sorodni korakom v prejšnjih dveh fazah:

- Izbira referenčnih modelov, zornih kotov in orodij: referenčnega modela nismo uporabili, temveč se omejujemo na elemente, ki podpirajo aplikacije iz informacijske arhitekture. Zaradi omejenega nabora ne bomo uporabili posebnih zornih kotov temveč predstavili celoten pogled in sicer z diagrami iz orodja Archi [20]
- Priprava opisa trenutne tehnološke arhitekture: slika 11-1: Trenutna tehnološka arhitektura (dekompozicijski diagram platform)
- Priprava opisa ciljne tehnološke arhitekture: slika 11-4: Ciljna tehnološka arhitektura (dekompozicijski diagram platform)
- Izdelava analize vrzeli med njima: slika 11-7: Analiza vrzeli tehnološke arhitekture
- Definicija kandidatov za komponente načrta: tu se pripravi grob načrt za prehod, ki ga nato natančneje definiramo v fazi E – Implementacija arhitekture. Sprejete odločitve so tako prikazane v poglavju 12 Implementacija arhitekture .
- Razrešitev posledic na celotno arhitekturo: nova tehnološka arhitektura ne bo imela večjih posledic na celotno infrastrukturo. Nove postavitve so namreč virtualizirani strežniki, za katere že imamo zmogljivosti.
- Formalni pregled z deležniki
- Finalizacija tehnološke arhitekture

- Dodajanje izdelkov v dokument arhitekturne definicije

Izdelki tehnološke arhitekture, ki nastanejo tekom naštetih korakov pa naslednji:

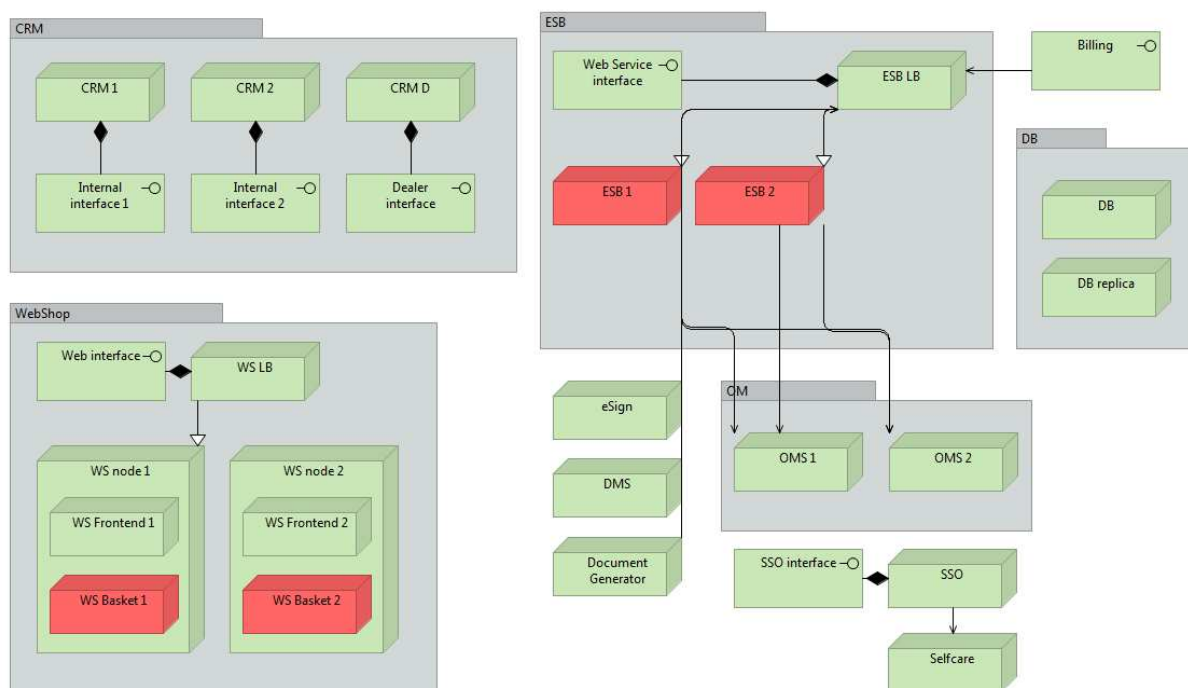
- Katalog tehnoloških standardov
- Katalog tehnološkega portfelja
- Matrika aplikacija/tehnologija
- Diagram okolij in lokacij
- Dekompozicijski diagram platform
- Diagram procesiranja
- Diagram mrežne opreme/strojne opreme
- Diagram komunikacij

Izmed predlaganih diagramov bomo za predstavitev tehnološke arhitekture izbrali dekompozicijski diagram platform, ki prikazuje logične in fizične komponente strojne in programske opreme. Z namenom povezovanja informacijske in tehnološke arhitekture bomo temu dodali še matriko aplikacija/tehnologija, vendar v diagramski različici. Spremembe v mrežni opremi in komunikacijah z zunanjim svetom v predstavljeni iteraciji življenjskega cikla poslovno-informacijske arhitekture ni, zato sta ta dva izdelka dovolj in lahko z njima pokrijemo vse relevantne spremembe, ki jih uvajamo.

## 11.1 Trenutna tehnološka arhitektura

Trenutna tehnološka arhitektura predvideva svoj sklop postavitvev za vse večje sisteme. Nekateri sklopi imajo več vozlišč, med katerimi je na različne načine porazdeljeno delo. CRM na primer ima tri instance. Prvi dve sta namenjeni interni uporabi, pri čemer uporabniki sami izberejo instanco, na kateri delajo, tretja pa je namenjena uporabi zunanjim pooblaščenim prodajalcem. Strežniki spletne trgovine in storitvenega vodila so skriti za t.i. »load balancerjem« oz. izenačevalnikom obremenitve, ki glede na obremenjenost posameznih vozlišč avtomatsko preusmerja zahteve. OM ima prav tako več vozlišč, vendar pridejo zahteve zanj v obliki sporočil, ki jih posamezna vozlišča naključno prevzemajo iz čakalne vrste in obravnavajo. Obračunski sistem je edini, ki ni lokalno postavljen, temveč se do njegovih storitev dostopa na oddaljeni lokaciji, podrobnosti njegove implementacije so zato skrite. Ostali sistemi so bodisi fizični ali virtualizirani strežniki na Windows ali Linux platformi. Kot smo že omenili v poglavju o informacijski arhitekturi, vsa komunikacija poteka preko storitvenega vodila (z izjemo relacije CRM - obračunski sistem). Kateri sistemi nato dejansko

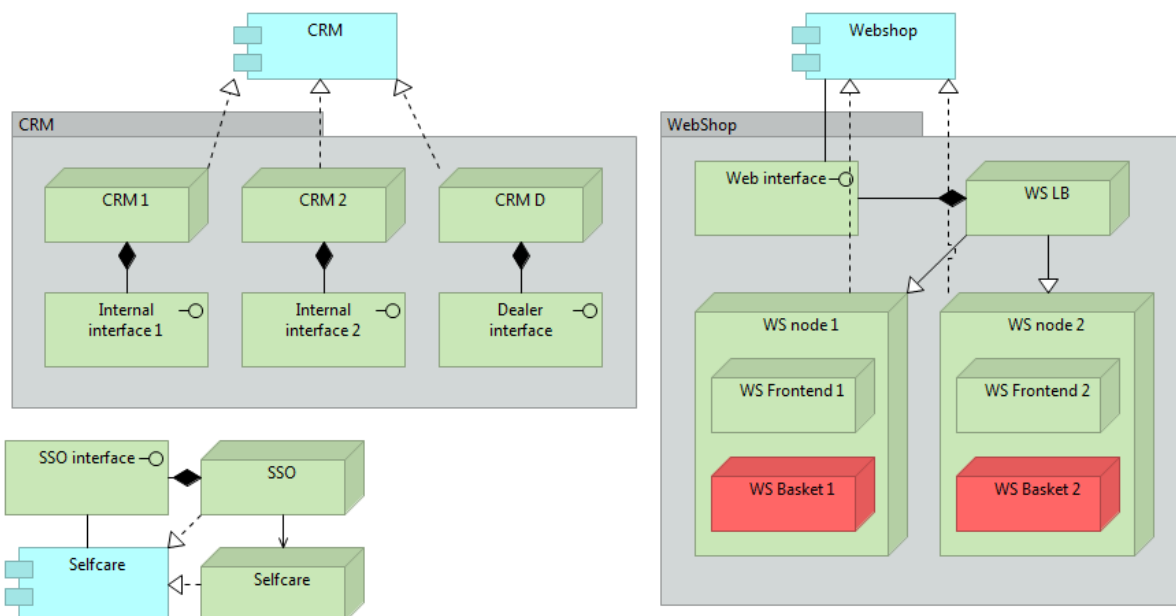
komunicirajo med seboj zato iz spodnje slike ni razvidno, so pa te iste povezave vidne v modelih aplikacijske arhitekture v poglavjih 10.1 Trenutna informacijska arhitektura in 10.2 Ciljna informacijska arhitektura. Prav tako niso narisane povezave do podatkovne baze. Do nje direktno dostopajo vsi sistemi z izjemo obračunskega sistema, ki ima svojo podatkovno bazo na svoji lokaciji.



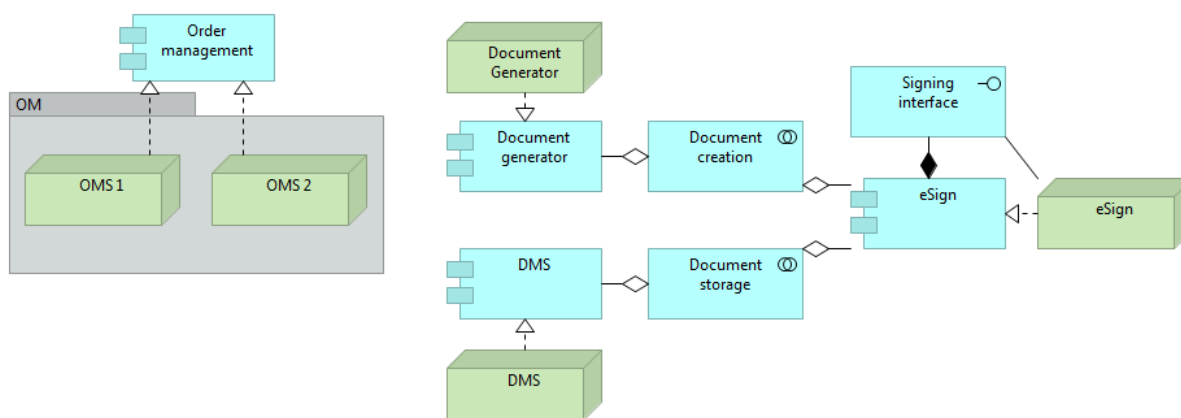
Slika 11-1: Trenutna tehnološka arhitektura (dekompozicijski diagram platform)

Takšen je torej celostni pogled na tehnološko arhitekturo samo. Za povezavo z informacijsko arhitekturo pa kot rečemo uporabimo diagrame povezav aplikacija/tehnologija, s katerimi lahko prikažemo, katere elemente tehnološke arhitekture uporablja posamezna aplikacija.

Na diagramih realizacije aplikacij so predstavljene postavitve sistemov, na katerih tečejo ključne BSS aplikacije. Med njimi ni obračunskega sistema, ki po trenutni arhitekturi ni postavljen lokalno in tak tudi ostaja. Njegova tehnološka postavitvev tako ni relevantna za našo arhitekturo.



Slika 11-2: Trenutna tehnična realizacija aplikacij - 1

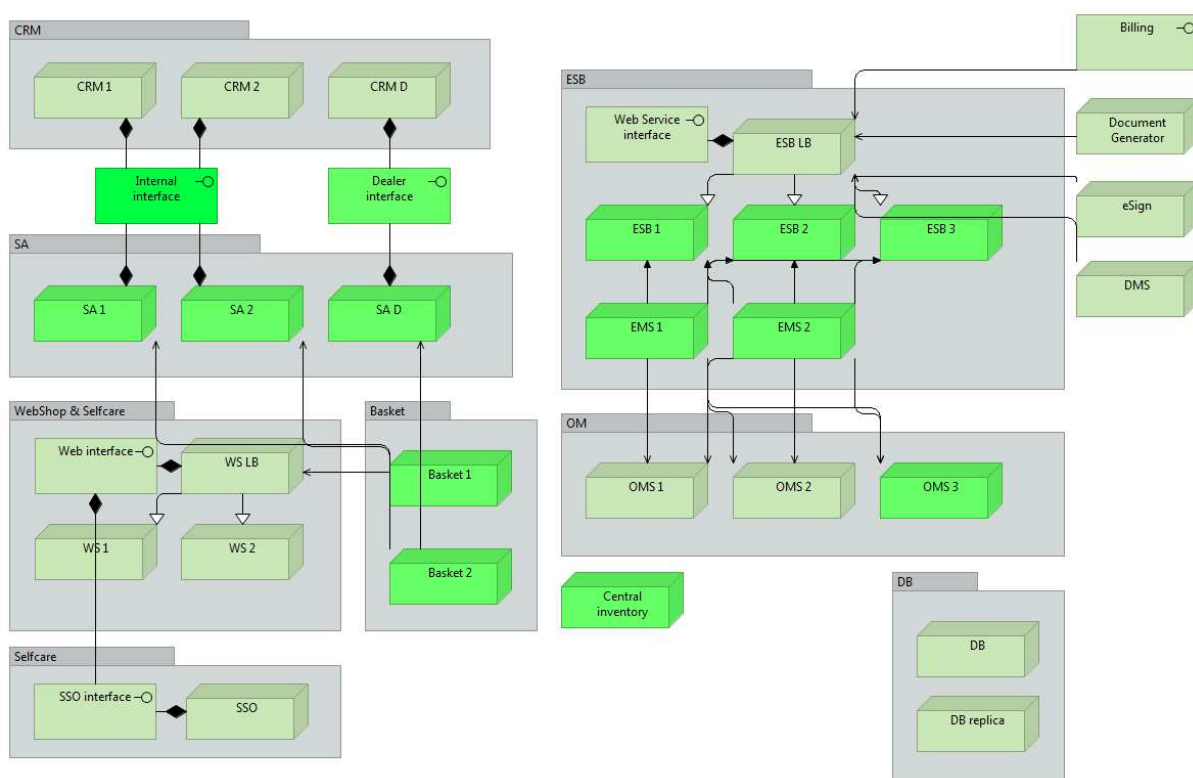


Slika 11-3: Trenutna tehnična realizacija aplikacij – 2

Kot vidimo so večji programski sklopi tudi realizirani z večjim številom tehnoloških elementov. To velja predvsem za CRM in spletno trgovino. Preostali manjši in manj kritični sistemi so realizirani po principu 1:1, torej en tehnološki element na eno aplikacijo. Pri njih je v ciljni tehnološki arhitekturi tudi najmanj sprememb.

## 11.2 Ciljna tehnološka arhitektura

Ciljna tehnološka arhitektura se bistveno ne razlikuje od trenutne. Vsi koncepti in načini komunikacije ostajajo enaki. Seveda se pojavljajo nove postavitve za na novo vpeljane aplikacije, nekaj pa je tudi odstranjenih vozlišč. V primeru OM se postavitve ohranja, vendar je število vozlišč povečano, saj nova informacijska arhitektura predvideva, da gredo vsa naročila čez ta sistem, kar bo prineslo dodatne obremenitve. Prav tako se zaradi predvidene večje obremenitve več postavitve storitvenega vodila (slika 11-4: Ciljna tehnološka arhitektura (dekompozicijski diagram platform)).

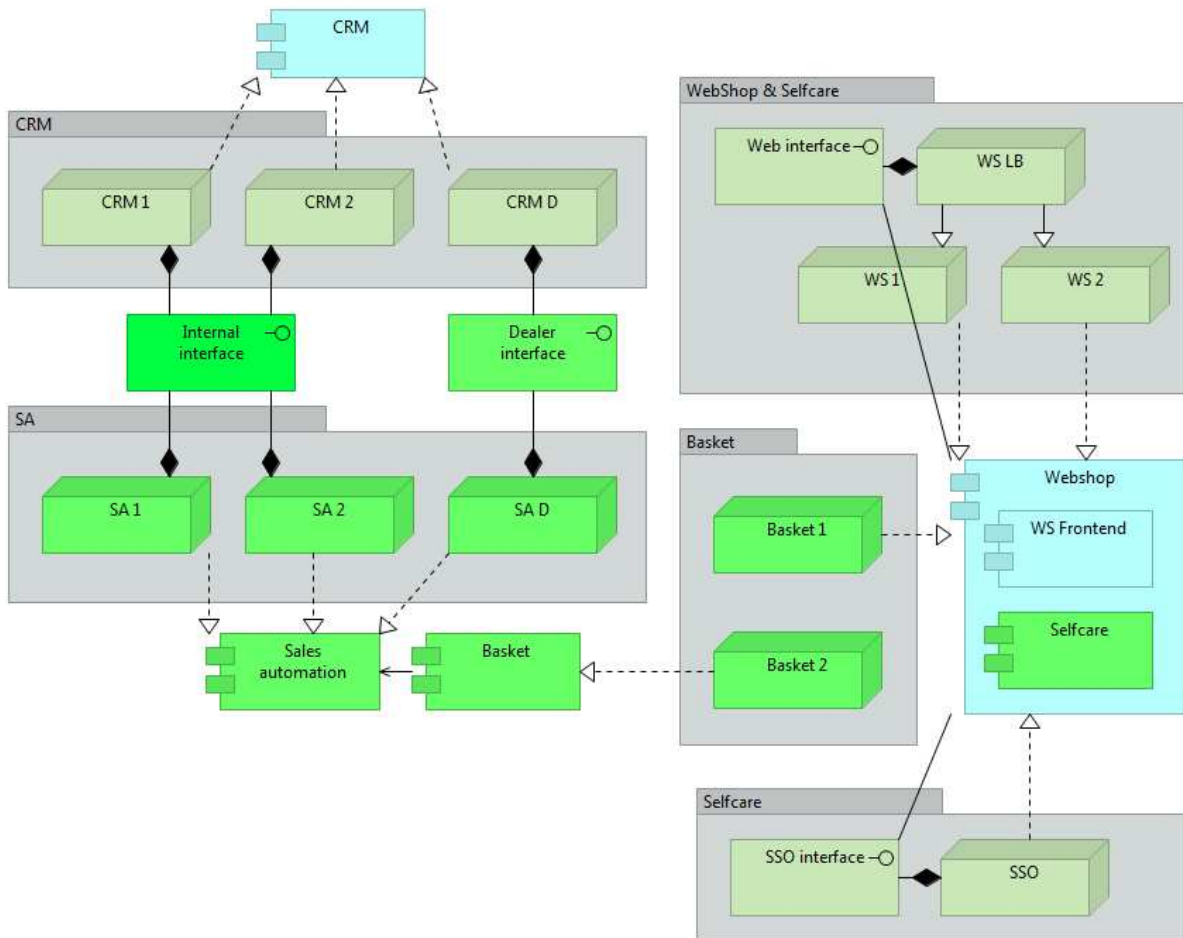


Slika 11-4: Ciljna tehnološka arhitektura (dekompozicijski diagram platform)

Realizacija aplikacij se v glavnem spreminja tam, kjer so se spremenile aplikacije same. Imamo torej novo strežniško postavitve za novo prodajno aplikacijo ter novo strežniško postavitve za novo verzijo nakupovalne košarice. Če je ta prej bila del spletne trgovine tako v aplikativnem kot strežniškem smislu, je sedaj to popolnoma samostojna komponenta z lastno postavitvijo.

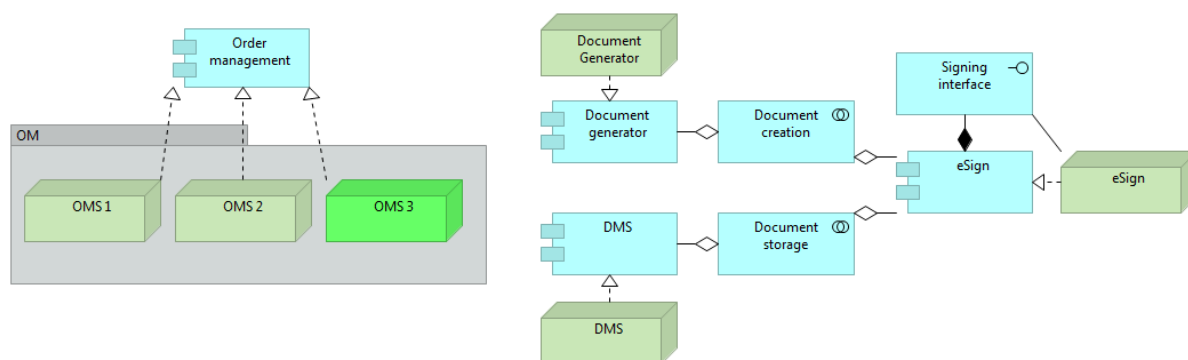
Kot rečeno za storitveno vodilo in OMS spreminjamo tehnološko postavitve v podporo večji obremenitvi, ne spreminjamo pa njenega koncepta in realizacije. Prav tako se realizacija

popolnoma nič ne spremeni za sisteme namenjene generiranju, podpisovanju in shranjevanju dokumentov.



Slika 11-5: Ciljna tehnološka realizacija aplikacij – 1

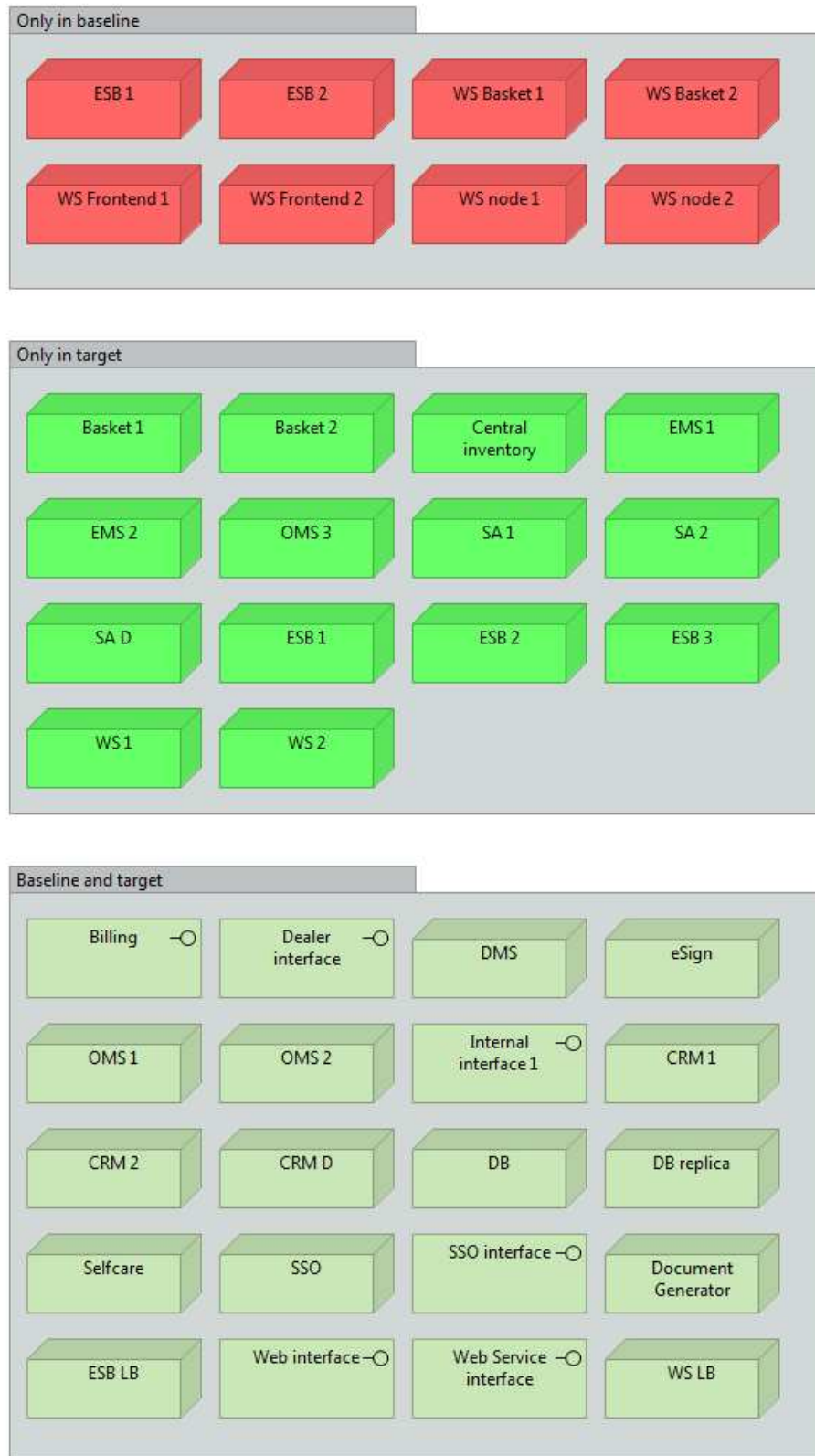




Slika 11-6: Ciljna realizacija aplikacij – 2

### 11.3 Analiza vrzeli tehnološke arhitekture

Čeprav se vsebinsko vloga CRM sistema zmanjšuje, njegova postavitvev ostaja nespremenjena. Vzporedno se ji doda enaka postavitvev prodajnega sistema (dve interni ter eno eksterno vozlišče), tako da vsak par tvori eno celoto. Postavitvev manjših podpornih sistemov ostaja nespremenjena, prav tako oddaljena postavitvev obračunskega sistema. Postavitvev spletne trgovine se z zunanjega vidika ohranja. Še vedno imamo eno vstopno točko v obliki izenačevalnika obremenitve, ki zahteve razporeja med dve vozlišči. Vsebina vozlišč pa je spremenjena zaradi integracije selfcare portala ter ukinitve spletne nakupovalne košarice. Kot smo že omenili se koncept postavitve OM ne spreminja, doda pa se novo vozlišče zaradi pričakovanega naraslega prometa, enako na področju storitvenega vodila. Popolnoma nova postavitvev je tudi strežnik za košarico. Ker je to kritična komponenta sta predvideni dve vozlišči. Eno bo namenjeno zahtevam s strani prodajnega sistema, drugo pa spletnemu kanalu. Med seboj se sinhronizirata preko enotne podatkovne baze. Centralni inventar prav tako potrebuje novo lastno postavitvev, zanj zaradi enotne komunikacije z OM-jem potrebujemo le eno aktivno vozlišče s pripadajočim rezervnim vozliščem, ki prevzame vlogo primarnega v primeru težav.



Slika 11-7: Analiza vrzeli tehnološke arhitekture

## 12 Implementacija arhitekture

Implementacija arhitekture pokriva tri faze: priložnosti in rešitve, plan migracije ter obvladovanje implementacije.

### 12.1 Faza E: priložnosti in rešitve

Namen faze E je priprava načrta za implementacijo ciljne arhitekture. Osnova zanj so analize vrzeli faz B, C in D, potrebno pa je tudi ugotoviti, ali je morda pri implementaciji potreben inkrementalni pristop, ki predvideva določeno število tranzicijskih arhitektur za lažji prehod od trenutne k ciljni arhitekturi.

Koraki faze E so sledeči:

- Potrditev ključnih korporativnih atributov sprememb: tu so mišljene kompetence vpletenih ter navade strank. Identificirani atributi so na primer:
  - delovni čas prodajnih mest: ponedeljek-sobota 8h-21h, nedelja 8h-15h
  - obiskanost spletnih kanalov: najmanjši obisk je med 3h in 4h uro zjutraj, največji pa med 10h in 12h ter med 16h in 19h
  - poznavanje procesov s strani zaposlenih: razmeroma dobro
  - sposobnost obravnavanja določenih procesov ročno: prodajni proces se lahko izvaja popolnoma ročno s kasnejšimi aktivacijami bodisi le v obračunskem sistemu ali po več zalednih sistemih
- Identifikacija poslovnih omejitev pri implementaciji: izpadi na prodajnih orodjih morajo biti omejeni na čas, ko prodajna mesta niso odprta, torej med 21h in 7h zjutraj oz. izjemoma ob nedeljah in praznikih po 15h, izpadi na spletnih orodjih morajo biti omejeni na ne več kot 3 ure in sicer nekje v času med 23h in 5h zjutraj zaradi delovanja klicnega centra in spletnih portalov. Oba kanala ne smeta imeti izpada hkrati (spletni in osebni). Druge poslovne omejitve so lahko na primer finančne narave, vendar take niso obravnavane v sklopu tega dela.

- Pregled in konsolidacija analize vrzeli iz faz B, C in D: V tem koraku se pregleda poglavja 9, 10 in 11 oz. predvsem 9.3, 10.3 in 11.3 ter naredi morebitne potrebne popravke za zadostitev ostalim zahtevam.
- Pregled konsolidiranih zahtev vseh poslovnih funkcij: v tem koraku se pregleda ali definirana ciljna arhitektura res izpolnjuje vse identificirane zahteve in to z minimalnim možnim naborom sprememb.
- Konsolidacija in uskladitev zahtev po interoperabilnosti: v tem koraku je potrebno pregledati vse nove točke integracije ter zagotoviti, da bo komunikacija med sistemi potekala nemoteno in sicer bodisi s pomočjo dodatnih vmesnikov ali s spremembami na sistemih samih. Konkretno potrebne spremembe bodo opisane v okviru tranzicijskih arhitektur.
- Validacija soodvisnosti: soodvisnosti so ključne pri postavitvi vrstnega reda del. V našem primeru povzročajo implementacijo v več korakih, tako da bo več o soodvisnostih v poglavjih namenjenih tranzicijskim arhitekturam.
- Potrditev pripravljenosti in tveganj za poslovno transformacijo: v tem koraku je potrebna potrditev vseh deležnikov na pripravljenost sprejetja tveganja ter pripravljenost na sodelovanje pri vseh potrebnih aktivnostih.
- Priprava implementacijske in migracijske strategije: potreba po korakih je že bila identificirana in je predstavljena v poglavjih o tranzicijskih arhitekturah.
- Identifikacijo in grupiranje večjih sklopov del: večji sklopi del so v našem primeru naslednji:
  - Vpeljava centralnega inventarja
  - Zamenjava dela CRM obrazcev s prodajnim orodjem
  - Uvedba generične košarice
  - Združevanje selfcare portala s spletno trgovino
- Identifikacija tranzicijskih arhitektur: tranzicijske arhitekture so posledica poslovnih omejitev. Da jim lahko zadostimo, ocenjujem, da sta potrebni dve tranzicijski arhitekturi, torej ciljno dosežemo šele s tretjo iteracijo sprememb. Več v poglavjih 12.1.1 in 12.1.2.
- Izdelava časovnice ter implementacijskega in migracijskega plana: tu je v okviru časovnice mišljen vrstni red del, ne pa tudi datumski okviri, ki so odvisni od razpoložljivih resursov in se določajo v fazi F. V našem primeru vrstni red določajo tranzicijske arhitekture, tako da več o vrstnem redu del v sledečih poglavjih.

### 12.1.1 Tranzicijska arhitektura 1

Kot že rečeno sta bili v okviru korakov faze E identificirani dve tranzicijski arhitekturi, ki sta potrebni za prehod na zahtevane ciljne arhitekture, upoštevajoč vse omejitve pri implementaciji

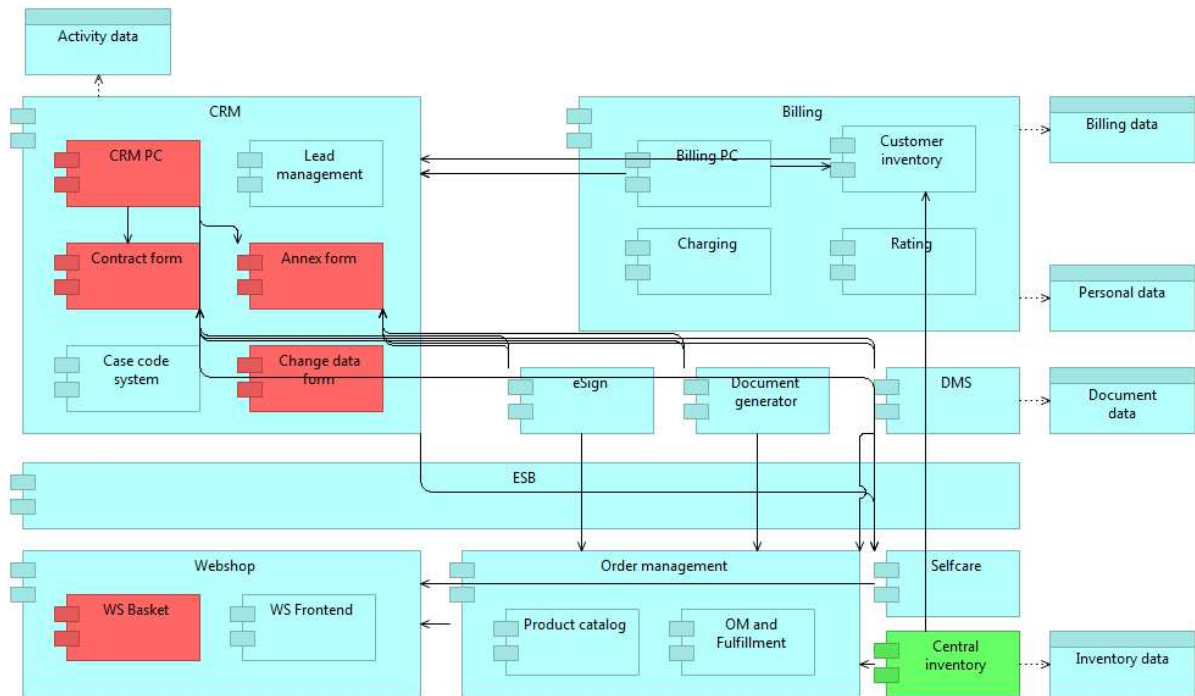
ter ob sprejemanju čim nižjega tveganja. Tako tveganja kot omejitve izvirajo iz informacijske arhitekture, zato začnemo z njo in nato definirani informacijski tranzicijski arhitekturi prilagodimo še poslovno in tehnološko.

#### 12.1.1.1 Informacijska tranzicijska arhitektura 1

V prvem koraku uvedemo centralni inventar. Le-ta je predpogoj tako za uvedbo prodajnega orodja kot za realizacijo sprememb in aneksov preko spletne trgovine. Oba namreč potrebujeata pregled komercialnih produktov, ki jih ima stranka, medtem ko inventar v trenutni arhitekturi predstavlja obračunski sistem, ki pozna le storitve, ne pa njihove komercialne predstavitve.

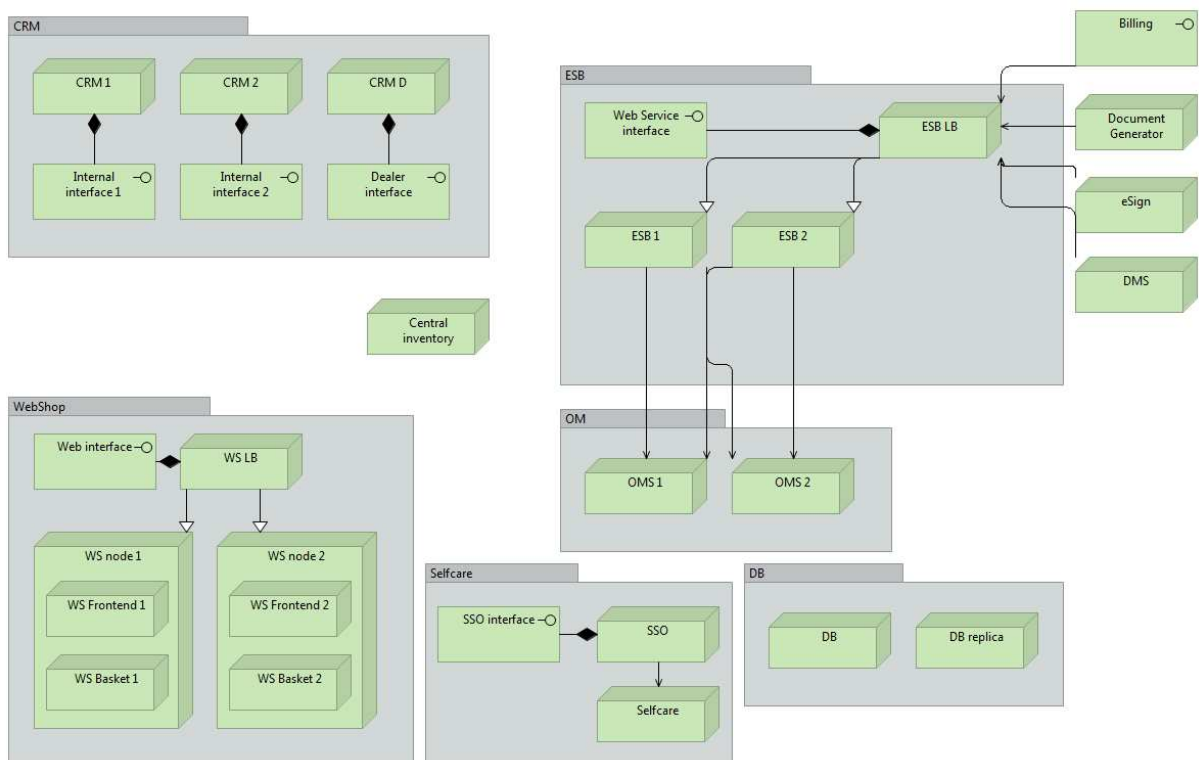
Poleg same uvedbe nove komponente, so potrebne sledeče prilagoditve (slika 12-1: Informacijska tranzicijska arhitektura 1):

- OM je potrebno dodati povezavo na centralni inventar ter sposobnost uporabe njegove vsebine
- V produktni katalog je potrebno dodati vse manjkajoče produkte in pripadajoča poslovna pravila
- Spletni trgovini je potrebno dodatni možnost avtentikacije za pridobivanje potrebnih podatkov o strankah preko OM
- OM and fulfillment komponenta mora vse nove produkte posebej nazaj zapisovati v centralni inventar
- Ker CRM še vedno uporablja star način dostave storitev, je potrebna dodatna sinhronizacija med obračunskim sistemom kot starim produktim inventarjem ter centralnim inventarjem kot novem primarnem viru inventarnih podatkov. Ta prilagoditev se lahko izvede na storitvenem vodilu z aplikacijo, ki ob vsakem vpisu v obračunski sistem, ekvivalentni vpis izvede še v centralni inventar, če ta še ni prisoten. To je le začasni ukrep, ki bo po prehodu na končno arhitekturo odstranjen, saj bo sistem za upravljanje z naročili kot edini skrbel za te zapise.



Slika 12-1: Informacijska tranzicijska arhitektura 1

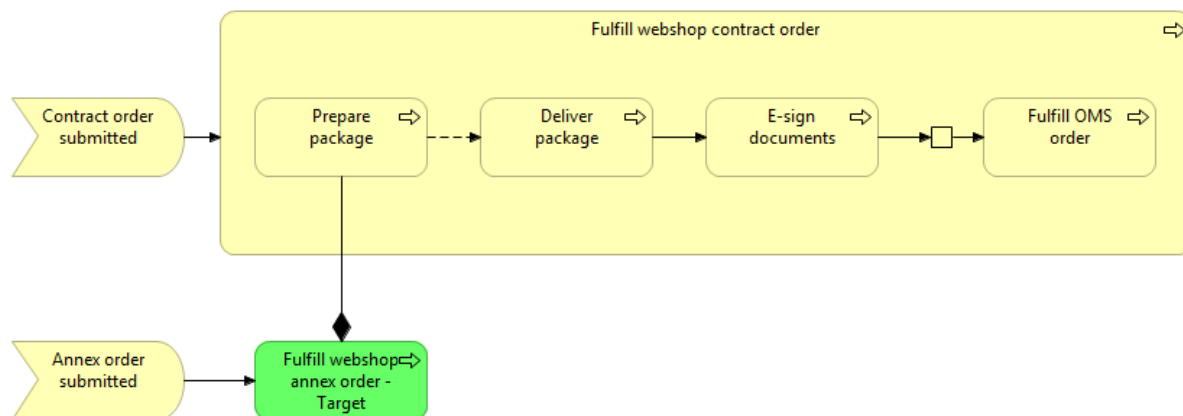
### 12.1.1.2 Infrastrukturna tranzicijska arhitektura 1



Slika 12-2: Infrastrukturna tranzicijska arhitektura 1

Infrastrukturna tranzicijska arhitektura sledi aplikacijski. Doda se postavitev strežnika, namenjenega centralnemu inventarju ter povezave med vsemi komponentami, omenjenimi v poglavju 12.1.1.1 Informacijska tranzicijska arhitektura 1.

### 12.1.1.3 Poslovna tranzicijska arhitektura 1



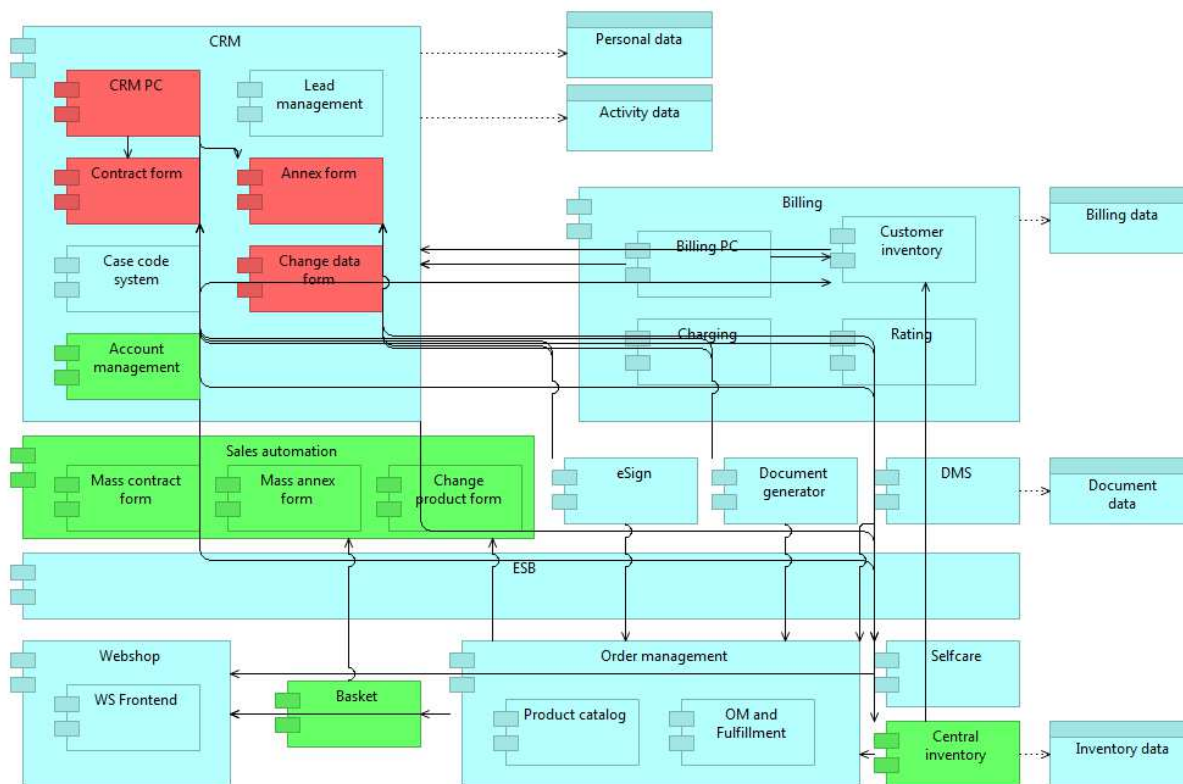
Slika 12-3: Poslovna tranzicijska arhitektura 1

Prva tranzicijska arhitektura na poslovnem področju prinaša sposobnost obravnavanja aneksov k pogodbi preko spletne trgovine brez potrebnega posredovanja s strani telefonske prodaje. Proces se v celoti lahko izvede avtomatizirano (slika 12-3: Poslovna tranzicijska arhitektura 1).

### 12.1.2 Tranzicijska arhitektura 2

Druga tranzicijska arhitektura prinaša večji sklop sprememb. Ponovno začnemo z aplikacijskim delom, ki narekuje tveganja in omejitve, ki jih dela prinašajo.

## 12.1.2.1 Informacijska tranzicijska arhitektura 2



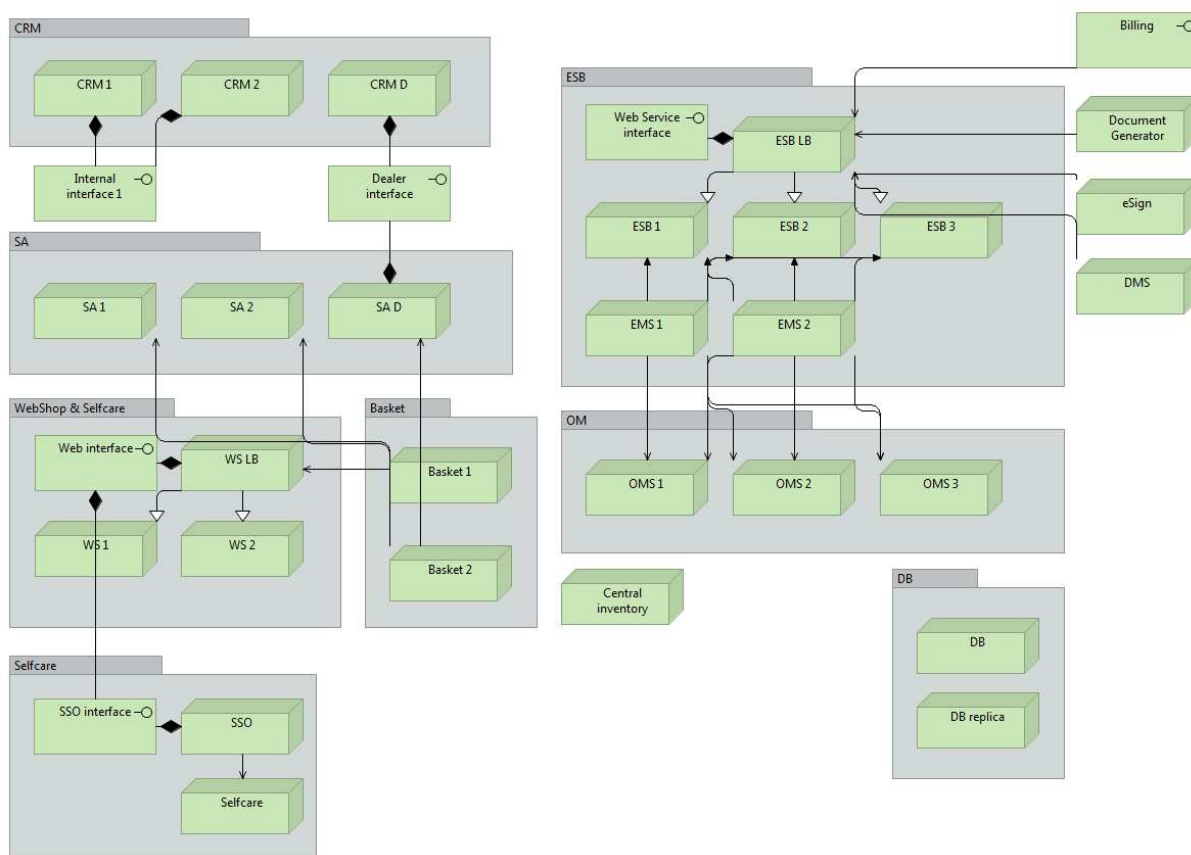
Slika 12-4: Informacijska tranzicijska arhitektura 2

V okviru druge tranzicijske arhitekture uvajamo prodajno orodje, generično košarico ter komponento za upravljanje z računi oz. strankami. Poleg postavitve samih novih komponent so potrebne sledeče prilagoditve:

- Uporaba generične košarice zahteva spremembe na spletni trgovini ter na načinu izvoza produktov iz produktnega kataloga v komponento košarice
- Zaradi izjemne kritičnosti CRM obrazcev, ki predstavljajo staro prodajno orodje, se le-ti v tej fazi še ne ukinjajo. Zaradi tega je v njih potrebna prilagoditev, da bodo sposobne uporabljati novo komponento za upravljanje s strankami. Prilagoditev potrebuje le komponenta za izdelavo pogodb ter za izvajanje sprememb.



### 12.1.2.2 Infrastruktura tranzicijska arhitektura 2



Slika 12-5: Infrastruktura tranzicijska arhitektura 2

Druga infrastruktura tranzicijska arhitektura tako kot tudi prva sledi aplikacijski. Postavijo se strežniki in povezave za vse nove komponente, prav tako pa se v tej fazi razširi kapacitete tako storitvenega vodila kot OM sistema (slika 12-5: Infrastruktura tranzicijska arhitektura 2).

### 12.1.2.3 Poslovna tranzicijska arhitektura 2

Z vidika poslovne arhitekture na tej stopnji ni sprememb. Procesi se ohranjajo, spremenijo se le orodja, s katerimi jih lahko izvajamo.

## 12.1.3 Prehod iz tranzicijske arhitekture 2 na ciljno arhitekturo

### 12.1.3.1 Prehod iz informacijske tranzicijske arhitekture 2 na ciljno informacijsko arhitekturo

V zadnjem koraku je potrebna le še ukinitve starih prodajnih obrazcev s popolnim preходом na novo prodajno orodje ter združitve selfcare portala s spletno trgovino.

Prvi del je razmeroma nezahteven. Preprosto je potrebno ugasniti komponente CRM-ja, ki so namenjene prodaji (CRM PC, Contract form, Annex form in Change data form). Le-te so ostale

v drugi tranzicijski arhitekturi le kot rezerva v primeru težav z novim prodajnim orodjem. Z ukinitvijo teh komponent se lahko odstrani tudi aplikacijo na storitvenem vodilu, ki smo jo uvedli v drugi tranzicijski arhitekturi in je služila sinhronizaciji inventarja. S temi koraki se končno preide na centralizirano upravljanje z naročili.

Drugi korak je združevanje selfcare portala s spletno trgovino. Njegov aplikacijski del se prestavi v sklop spletne trgovine, vmesnik pa se združi z vmesnikom spletne trgovine, zaradi česar so na njem potrebne prilagoditve.

To je še zadnji potreben korak migracije in z njim dosežemo končno ciljno arhitekturo.

#### 12.1.3.2 Prehod iz infrastrukturne tranzicijske arhitekture 2 na ciljno infrastrukturno arhitekturo

Ukinitev določenih komponent CRM-ja nima vpliva na fizično postavitve njegovih strežnikov, tako da tukaj z infrastrukturnega vidika ni sprememb.

Spremembe pa so na strani selfcare portala. Njegov strežnik se ukinja, aplikacijska komponenta pa se prestavi na WS1 in WS2, vozlišči spletne trgovine.

#### 12.1.3.3 Prehod iz poslovne tranzicijske arhitekture 2 na ciljno poslovno arhitekturo

S poslovnega vidika v procesih tekom tega zadnjega koraka ni sprememb. Vsebinsko se za vpletene akterje spremeni le to, da za prodajo nimajo več dveh orodij, temveč le enega, ter da uporabniki vse svoje spletne procese izvajajo preko enega vmesnika namesto dveh.

## 12.2 Faza F: Planiranje migracije

Namen faze F je pripraviti natančnejši migracijski plan na osnovi ugotovitev iz faze E. Tukaj se predvsem dodatno upošteva potrebne resurse in njihov vpliv na časovnico, ter se skladno z možnostmi organizacije pripravi tudi časovni plan migracije z mejniki, roki in natančno definiranimi odgovornimi osebami.

Koraki faze F so torej sledeči:

- Potrditev interakcij implementacijskega in migracijskega plana z menedžerskimi ogroddji
- Dodelitev poslovne vrednosti vsakemu delovnemu sklopu
- Ocena potrebnih resursov in časovnih okvirov oz. rokov

- Prioritizacija migracijskih projektov skozi oceno stroški/koristi
- Potrditev arhitekturne časovnice ter posodobitev dokumenta arhitekturne definicije
- Izpeljava implementacijskega in migracijskega plana
- Zaključek cikla razvoja arhitekture ter dokumentacija poteka

Koraki faze F zahtevajo konkretno organizacijo s konkretno definiranimi resursi, zato je v sklopu tega dela ne bomo obravnavali.

## 12.3 Faza G: Upravljanje implementacije

Namen faze G je dejanska izvedba migracijskega plana iz faze F s primernim upravljanjem implementacije in nadzorom vpliva na poslovanje ter pregledom zahtev po spremembah.

Koraki faze G:

- Potrditev obsega in prioritet namestitve
- Identifikacija namestitvenih resursov in njihovih kompetenc
- Priprava navodil za namestitev rešitev
- Pregled skladnosti rešitev s poslovno-informacijsko arhitekturo)
- Implementacija poslovne in tehnične podpore storitvam
- Pregled implementacije ter zaključek aktivnosti

Koraki faze G prav tako zahtevajo konkretno organizacijo s konkretno definiranimi resursi, zato je v sklopu tega dela ne bomo obravnavali.

S to fazo se postavitev nove arhitekture zaključí in je cilj za ta življenjski cikel dosežen.



## 13 Upravljanje sprememb v arhitekturi

Kot že rečeno je faza H namenjena vzdrževanju postavljene arhitekture in obravnava morebitne spremembe na arhitekturi, ki se lahko pojavijo zaradi prilagoditve poslovanja, reorganizacije, izteka življenjske dobe strojne in programske opreme in podobnega.

Spremembe se torej dogajajo kot posledica določenega dogodka, nezadovoljstva oz. morebitnih napak iz naslova sprememb v tekočem ciklu. Takih primerov v sklopu tega dela ne bomo obravnavali.



## 14 Sklepne ugotovitve

V današnjem hitro spreminjajočem se poslovnem svetu je podpora informacijske tehnologije ključna za uspeh. Predvsem pa je ključna njena sposobnost prilagajanja tako trgu kot organizaciji sami. Poslovno-informacijska arhitektura je veda, katere namen je učinkovito sledenje in prilagajanje takim spremembam. Obsega tako procesni vidik kot aplikacijski in strukturni vidik in organizacijam pomaga ob prilagoditvah na dogodke kot so spremembe organizacijske strukture, procesne spremembe, zamenjava strojne in programske opreme zaradi izteka življenjske dobe, odzivi razmeram na trgu, širitve poslovanja, združevanja ter odkupi drugih organizacij in podobno.

Telekomunikacije so ena izmed panog, ki so še posebej podvržene odvisnosti od informacijske tehnologije. Praktično vse storitve in produkti, ki jih nudi, temeljijo na določeni tehnični implementaciji. Spremembe v postavitvi in uporabi le-te so zato tako rekoč vsakodnevna stalnica. Včasih pa je kljub konstantnemu spreminjanju potrebna tudi bolj temeljita prenova. Konkretno smo se v primeru tega dela lotili prenove dela informacijske tehnologije, ki je namenjen podpori poslovanju v telekomunikacijskem podjetju.

Prenovo je zahtevala dotrajanost stare programske opreme, ki ni bila več kos sodobnim zahtevam po vse hitrejši implementaciji novih in raznolikih storitev ter je zahtevala visoke stroške obratovanja. Ker gre za sisteme, ki so kritični za poslovanje in imajo tudi velik vpliv na poslovne procese, je bil potreben celovit pogled na prenovu. Od tod tudi odločitev za uporabo ogrodja TOGAF in njegove metodologije ADM, s pomočjo katere je bila določitev nove arhitekture bolj sistematična, predvsem pa je upoštevala vse vidike poslovanja. Upoštevali smo razloge za prenovu, organizacijske cilje in vrednote, ki so temelj za usmeritev celotnega življenjskega cikla arhitekture, ter pripravili definicije arhitekture same z vidika poslovanja, aplikacij, podatkov in infrastrukture. Vse prevečkrat se ob tovrstnih projektih namreč dogaja, da se izhaja iz enostranskega pogleda. Na primer, da se poskuša eno zastarelo aplikacijo preprosto nadomestiti z drugo, ki mogoče prinese kakšne dodatne funkcionalnosti, ne rešuje pa globljih težav in ne upošteva vpliva na poslovne procese.

Ključni faktor pri predstavitvi idej in zaključkov celotne metodologije je bila tudi odločitev po uporabi jezika ArchiMate, s katerim so vsi vidiki arhitekture konsistentno in nazorno predstavljeni.

Rezultat je tako pregleden in celovit popis vseh vidikov poslovno-informacijske arhitekture s pripadajočimi ciljnim arhitekturami in migracijskim planom, kar prinaša ne le kvalitetnejše odločitve glede posameznih korakov v življenjskem ciklu arhitekture temveč tudi uspešnejše načrtovanje in realizacijo prehoda na izbrano arhitekturo ter spremljanje in prilagajanje izbrane arhitekture, ko je že v uporabi.

V nadaljevanju bi veljalo uporabljen princip razširiti še na raziskavo o modeliranju vse treh tipov arhitektur tudi na nižjih nivojih. Definirati bi bilo potrebno, na kateri stopnji bi bilo smiselno zapustiti svet poslovno-informacijske arhitekture, TOGAF-a in ArchiMate modeliranja ter kakšno prakso, metode in jezike uporabiti na nižjih nivojih. S tem bi pridobili celotno paleto znanj in smernic za izvedbo projekta celostne prenove arhitekture.



# Viri in literatura

## Literatura

- [1] Bub U., Picot A., Krcmar H., The future of telecommunications. *Business & Information Systems Engineering* 3(5):265–267, 2011
- [2] David Chen, Guy Doumeingts, Francois Vernadat, Architectures for enterprise integration and interoperability: Past, present and future, *Computers in Industry* 59 (2008) 647–659
- [3] Wilco Engelsman , Dick Quartel , Henk Jonkers & Marten van Sinderen, (2011) Extending enterprise architecture modelling with business goals and requirements, *Enterprise Information Systems*, 5:1, 9-36
- [4] IEEE 1471, Recommended Practice for Architectural Description of Software-Intensive Systems, 2000.
- [5] ISO 15704, Industrial Automation Systems—Requirements for Enterprise reference Architectures and Methodologies, 2000.
- [6] Henk Jonkers (Ed.), Iver Band, Dick Quartel, Henry Franken, Mick Adams, Peter Haviland, Erik Proper, Using the TOGAF® 9.1 Architecture Content Framework with the ArchiMate® 2.0 Modeling Language , The Open Group, July 2012
- [7] Henk Jonkers, Marc Lankhorst, René Van Buuren, Stijn Hoppenbrouwers, Marcello Bonsangue, Leendert Van Der Torre, Concepts for modeling enterprise architectures, *International Journal of Cooperative Information Systems* Vol. 13, No. 3 (2004) 257–287, World Scientific Publishing Company
- [8] Henk Jonkers, Marc M. Lankhorst, Hugo W.L. ter Doest, Farhad Arbab, Hans Bosma, Roel J. Wieringa, Enterprise architecture: Management tool and blueprint for the organisation, *Information Systems Frontiers*, February 2006, Volume 8, Issue 2, pp 63-66
- [9] Marc Lankhorst et al. , Enterprise Architecture at Work, Modelling, Communication and Analysis, Second Edition, Springer 2009

- [10] The Open Group, ArchiMate® 2.1 Specification, Second edition, second impression, May 2015, Van Haren Publishing, Zaltbommel
- [11] The Open Group, TOGAF® Version 9.1, 2011
- [12] Ross, J.W., Creating a strategic IT architecture competency: learning in stages. MIS Quarterly, 2003, (2) 1, 31 - 43.
- [13] Ross J.W., Weill P., Robertson D.C., Enterprise architecture as strategy: Creating a foundation for business execution, Harvard Business Press 2006, EUA
- [14] Ana Šaša, Marjan Krisper, Enterprise architecture patterns for business process support analysis, The Journal of Systems and Software 84 (2011) 1480–1506
- [15] Business Process Framework, GB921 Addendum DX, Release 15.5.1, TM Forum, Marec 2016

## Ostali viri

- [16] (2013) Gartner IT Glossary – Enterprise Architecture (EA). Dostopno na: <http://www.gartner.com/it-glossary/enterprise-architecture-ea/>
- [17] (2015) ArchiMate - Wikipedia, the free encyclopedia. Dostopno na: <https://en.wikipedia.org/wiki/ArchiMate>
- [18] (2015) The Open Group Architecture Framework - Wikipedia, the free encyclopedia. Dostopno na: [https://en.wikipedia.org/wiki/The\\_Open\\_Group\\_Architecture\\_Framework](https://en.wikipedia.org/wiki/The_Open_Group_Architecture_Framework)
- [19] (2010) The Contribution of Enterprise Architecture to the Achievement of Organizational Goals: Establishing the Enterprise Architecture Benefits Framework, Technical Report, Department of Information and Computing Sciences Utrecht University, Utrecht, The Netherlands. Dostopno na <http://www.cs.uu.nl/research/techreps/repo/CS-2010/2010-014.pdf>
- [20] (2016) Orodje Archi. Dostopno na <http://www.archimatetool.com/download>